Viz Pilot User Guide

Version 8.9

Contents

1   Introduction

Welcome to the Viz Pilot User Guide.

Create, manage and deliver high volumes of top-quality content to live and prerecorded news, sports, election and other broadcast productions, independently of the design department. The Viz Pilot system is the core content control system for broadcasters who require speed, a consistent look, and a streamlined workflow for their graphics and video content.
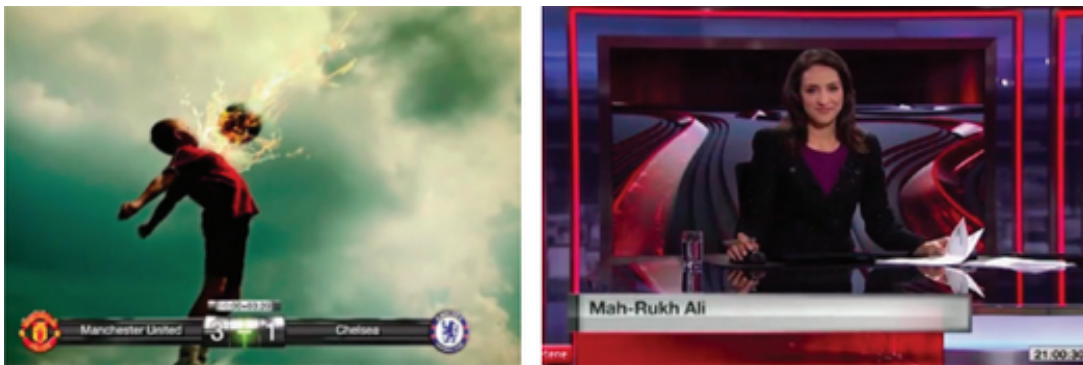
1.1   Key Features

- **Content Creation:** A unique and easy solution for graphics and video content creation. Data can be entered either directly into Director, or through Vizrt's Newsroom Integration using Viz Pilot News. Data can then be added to a playlist for playout automation.
- **Playout Automation:** The Viz Pilot playlist gives you seamless playout of graphics and video. Playlists are typically created in newsroom systems or Director, and can be monitored and played out using Director or

another third party control application. Elements can be triggered in Director, through GPI or third-party integrations.

> ⚠️ **Note:** For a complete overview of the newest features and bug fixes, see the release notes.

1.2   Feedback

We welcome your feedback on Vizrt products and documentation. Please contact your local Vizrt customer support team at http://www.vizrt.com.

2 Viz Pilot Components

Viz Pilot consists of multiple applications:

- Viz Pilot News
- Director
- Template Wizard
- Viz Pilot database
- Pilot Data Server
- Object Store (image database)
- Thumbnail Generator

2.1 Pilot Data Server

Pilot Data Server acts as a server for accessing Viz Pilot's database and other services, and must be installed in order to use a number of features such as Crop Service, Template Tagging, Update Service, Person Search, search on Viz One, Thumbnail Generator and Timeline Editor. It is also required when using Graphic Hub as the Viz Pilot database, or for integrating with Viz Story.

2.2 Director

Director is dependent on several systems in order to work in a control room, the most important of which are Media Sequencer and the Viz Pilot database. The basic setup therefore consists of a connection to Media Sequencer and the Viz Pilot database. A Viz Engine with a connection to Graphic Hub is also required for playout. Any graphics or images added to a playlist must be available to Viz Engine.

A common setup is also to run Director with a newsroom system connection; this allows Director to retrieve playlists made available by the newsroom system. The Director operator will then only monitor and control playout of the playlist elements, and not create them.

A connection to Viz One for the search and transfer of video clips to Viz Engine can also be used for playout of video clips.

> ⚠️ **Note:** Media Sequencer is used to configure sequencer connections to the Pilot system, see Media Sequencer Configuration. The Media Sequencer may be installed locally or on a separate server (see Command Line Startup Options).

2.3 Overview Of Components

Type	Description
Director	The main application for managing graphics and video playout of both MOS rundowns and any other playlist. The user has full control over templates, data elements and content. It is possible to manually edit and play out all media available in the Viz Pilot system, and play out scenes from Graphic Hub. ⚠ **Note:** Although the director installer allows for a --migrate switch that copies the configuration from previously installed versions of director when upgrading, this process only works if the previous version is still installed.
GPI drivers	General Purpose Input (GPI) drivers permit external control of Viz Pilot.
Local preview plugin	Used for previewing scenes locally in Director and Viz Pilot News. Note that it is recommended to use the Remote Preview or Remote Snapshot Preview rather than this component. See Viz Engine Preview Configuration.
Local documentation	Local documentation for the Viz Pilot system.
Object Store	Helps store and retrieve images, sound and video clips to and from the file share (stills and person information), and the Viz Pilot database (thumbnails), for use in templates.
Oracle Drivers	Oracle Instant Client is used for accessing an Oracle database, if applicable to your configuration.
Quick CG	A system tray application for the quick creation of character generated (CG) data elements.
Spellcheck plugin	Installs a set of default OpenOffice dictionaries that use the Hunspell spell checker. Includes Arabic (both North African and Middle Eastern), English (both UK and US), French, German, Norwegian (both Bokmaal and Nynorsk), Russian and Spanish. See OpenOffice.org. Dictionaries are UTF-8 formatted by Vizrt.
Template Wizard	Creates templates to use within the Viz Pilot system.
Viz Pilot News	Integrates into the newsroom system allowing journalists to add graphics to a story.

3   Software Requirements

> ⚠ **Note:** This is a general overview only. Always check the **release notes** for exact requirements.

This section contains the following topics:

- Viz Pilot System Requirements
- Viz Pilot Minimum Software Requirements
- Director Requirements
- Viz Pilot News Requirements
- Template Wizard Requirements
- Pilot Data Server Requirements
- Crop Service Requirements
- Viz Engine Preview Server Requirements

3.1   Viz Pilot System Requirements

These are the minimum requirements for Viz Pilot:

Component	Minimum Requirement
System	- Microsoft Windows 10 Enterprise (64-bit) - Microsoft Windows 10 Pro (64-bit) - Microsoft Windows Server™ 2016 (64-bit) - Microsoft Windows Server™ 2019 (64-bit) ⚠ **Note:** Certain specific operating system versions and service packs are not supported after the official Microsoft support end date, which is also the end date for Vizrt support. End dates for support that expires before the official Microsoft support end date will be announced in release notes for new versions of Viz Pilot.
RAM	2 GB RAM (64-bit)
HDD	40 GB hard drive with at least 15 GB of available space.

3.2   Viz Pilot Minimum Software Requirements

This section details the software requirements for Viz Pilot:

Software	Version
Graphic Hub	3.6.0 (with Graphic Hub REST 2.6.0) or Oracle 11.
Media Sequencer	5.4.0 (Media Sequencer 5.5.0 for communication over HTTPS with TLS 1.2)
Viz Engine	4.4.1
Viz World Client	17.0.1 The integration might work with earlier versions as well, but not without issues.
Pilot Data Server	8.9 The integration should work with earlier versions, but this has not been tested.

⚠ **Note:** Irfan View is required for enabling preview of some templates when running Windows 10.

3.2.1   Viz Pilot Database - Oracle

In this case, the Viz Pilot database is an Oracle database server. Usually, two database servers are installed where one is used for manual failover. Viz Engine graphics data is stored on Graphic Hub.

The Viz Pilot Database Administrator tool (Pilot DBA) is a small application used for installing, upgrading, exporting, importing, and setting various parameters for the Viz Pilot Database. It is purely a DBA tool, and should therefore only be used by database administrators.

3.2.2   Viz Pilot Database - Graphic Hub

In the current example, the Viz Pilot database is Graphic Hub, accessed via a Graphic Hub REST server. For full installation and migration information, see the Graphic Hub User Guide.

Migration from Oracle to Graphic Hub

To move data from Oracle to Graphic Hub, the schema in the Oracle database must be Viz Pilot 6.x or later. To upgrade the schema used by the Pilot DBA, see Database Administration. Once the schema is 6.x or later, import the data into Graphic Hub.

Once the data has been imported into Graphic Hub, the schema version will follow the Graphic Hub version. Any required schema updates will be made the first time Graphic Hub is started up after a version upgrade for the selected data directory.

⚠ **Note:** The Pilot Database Administrator tool (Pilot DBA) is no longer required once you have migrated to Graphic Hub.

New Installations

For new installations, install Graphic Hub as described in the Graphic Hub User Guide.

3.3   Director Requirements

The table below lists machine specifications for a typical newsroom system setup, where Director is the control application receiving the playlist:

Component	Minimum Requirement
Software	• Viz Pilot • Media Sequencer ⚠️ **Note:** In order to configure the MSE database in Director, it is required to have installed VS13 libraries. They can be found here: https://www.microsoft.com/en-us/download/details.aspx?id=40784. *Optional*: Viz World Client (32-bit), TimeCode Monitor, Viz PreCut. *Optional:* If you are using a local Viz Engine to preview video clips from Viz One, Viz Engine must be installed with video codecs (for example MPEG-4 codec and Haali Media Splitter). *Optional:* Windows Media Player 11 for video clip preview in Object Store.
Local Drive Access	Oracle database is used to access Oracle client files and folders.
Network Access	Mapped drive to Object Store still store folder.

ℹ️ **Info:** Director receives its Viz One connection parameters from the Viz Pilot database.

⚠️ **Note:** Using remote preview that does not require a local Viz Engine or graphics cards on the client machine is recommended.

3.4   Viz Pilot News Requirements

The table below shows a basic setup of Viz Pilot News:

Component	Requirement
Software	• Viz Pilot *Optional*: Object Store Viz World Client (32-bit) 12.0 or later • Newsroom System Client *Optional*: If you are using a local Viz Engine to preview video clips from Viz One, Viz Engine must be installed with video codecs (for example MPEG-4 codec and Haali Media Splitter). *Optional:* Viz EasyCut or Viz PreCut for video clip editing.
Local Drive Access	Read access to Oracle client files and folders, if using Oracle.
Network Access	Mapped drive to Object Store still store folder.

> ⚠️ **Note:** Using remote preview that does not require a local Viz Engine on the client machine is recommended. Local preview is not recommended.

> ℹ️ **Info:** Viz Pilot News receives its Viz One connection parameters from the Viz Pilot database.

> ⚠️ **Note:** The timeline editor in Viz Pilot News does not work on virtual machines.

3.5   Template Wizard Requirements

Template Wizard can be installed as a stand-alone product for editing and running Viz Pilot templates without a Viz Pilot Database server; for example, for Viz Trio or Viz Ticker3D. No Oracle Client is required for stand-alone installation. The `use_database` setting in the ContentPilot.ini file (under the [TEMPLATE_WIZARD] section) must be set to 'n'.

Although it can be installed on the same machine as Viz Artist, it's recommended to install Template Wizard on a separate design client for template designers.

3.5.1   Template Wizard Specifications

Component	Requirement
Software	*Optional:* Media Sequencer *Optional*: Microsoft Windows Media Player 11 for video clip preview in Object Store. *Optional*: Viz World Client (32-bit)

Template Wizard connects to the Media Sequencer for testing and previewing template graphics. The Media Sequencer connection defaults to *localhost*. To use a different host, set the command line option *-mse <host>* in Template Wizard's target path.

3.6   Pilot Data Server Requirements

3.6.1   Pilot Data Server

The Pilot Data Server is installed as an application layer on top of the Viz Pilot database. It acts as an application server for accessing the Viz Pilot database and other services. The Pilot Data Server may be asked to handle requests from scripts to provide information on data elements, or to tell Preview Server which scenes and data to render.

The Script Runner Service comes packaged with the Pilot Data Server. Thumbnail Generator is an optional component, which can be selected during Pilot Data Server installation.

3.6.2   Script Runner Service

The Script Runner Service provides users with a simple way to use the Update Service without needing to create their own service. See Update Script Editor.

3.6.3   Thumbnail Generator

The Thumbnail Generator is a Windows service that automatically generates thumbnails of data elements as they are created or changed. The thumbnails are stored in the Viz Pilot database.

Thumbnail Generator requires a Preview Server, which is used to serve the thumbnails in the data element feed.

3.6.4   Pilot Data Server Specifications

Component	Requirement
Software	Pilot Data Server
Services	Vizrt Pilot Data Server Vizrt Script Runner Vizrt Thumbnail Generator
System	See Viz Pilot Release Notes

3.7   Crop Service Requirements

3.7.1   Crop Service

Crop Service is a tool for basic cutting and zooming of images, which allows users to quickly add new images to a data element.

3.7.2   Crop Service Specifications

Component	Requirement
Software	Crop Service
Services	Vizrt Crop Service
System	See the Viz Pilot Release Notes

3.8   Viz Engine Preview Server Requirements

The Preview Server is used in situations where one or more Viz Engines provide frames for snapshot or thumbnail generation. It is used by the Viz Trio client or Viz Pilot to fetch previews of overlay graphics; for example, for the Timeline Editor or the Thumbnail Generator.

> ⚠ **Note:** Although the Preview Server can run on the same machine as a Viz Engine, it's also possible to specify additional Viz Engines in order to spread the load.

Preview Server can manage a pool of Viz Engines, and clients can query it for a Viz Engine that matches a specific video mode. Preview Server provides load balancing and video mode matching, and can be used to create a redundant renderer pool. The pool of Viz Engines is configured on the Preview Server Web Interface.

For complete information on requirements, see the *Preview Server Administrator Guide*.

4   Installation

The latest official releases of all Vizrt software are available on Vizrt's FTP server.

This section covers the following topics:

- Newsroom Integration Installation
- Control Room Client Installation
- Template Design Client Installation
- Video Codecs
- Crop Service Installation
- Pilot Data Server Installation
- Viz Pilot System Installation
- Unattended Installation

4.1   Newsroom Integration Installation

This installation type applies to a newsroom work station, where Viz Pilot News is integrated with one or more newsroom or NLE systems. The software is normally integrated to newsroom systems as an embedded component. This type of installation must be used when using Viz Pilot News together with Vizrt's Graphics Plugin for NLE.

> ⚠️ **Note:** When installing Viz Pilot News to use in a Java or web-based newsroom system, the Oracle Instant Client should _not_ be installed.

> ⚠️ **Note:** As most newsroom computers running it do not have administrator rights, you may need to install Viz Pilot News using **RunAs** parameters or install the software from the command line. You must have administrator rights for this to work.

> ⚠️ **Note:** In order to use the *NewsroomConnectTestPage* on systems where Internet Explorer 11 is not available, copy the test page to a web server (such as the Pilot Data Server). Then configure Microsoft's Enterprise Site List to load this URL in IE8Enterprise mode. Use Microsoft Edge to visit the page on the new URL to load the Pilot News plugin.

The Viz Pilot newsroom installation includes the following Viz Pilot Components:

- Viz Pilot News
- Object Store
- Documentation
- Oracle Drivers
- Local preview plugin
- Spell check plugin
- Quick CG

> ⚠️ **Note:** The Local preview plugin can be removed if you prefer a setup that uses a remote Viz Engine for preview. The Oracle Drivers can also be removed if an instance of the Oracle Instant Client has already been installed.

See Also

- Viz Pilot System Installation
- Configuring newsroom settings

4.2   Control Room Client Installation

The Viz Pilot control room installation manages playlists for playing out graphics on air and includes the following Viz Pilot Components:

- Director
- Object Store
- Quick CG
- Spell Check Plugin
- Template Wizard

> ⚠️ **Note:** Director must have a Media Sequencer running. It's recommended that Media Sequencer be run on the same machine.

See Also

- Viz Pilot System Installation

4.3   Template Design Client Installation

This installation type applies to a Template Wizard work station, where template designers create and manage the graphics templates.

The Template Wizard installation includes the following Viz Pilot Components:

- Template Wizard
- Director
- Object Store
- Documentation
- Oracle Drivers
- Spell Check Plugin

See Also

- Viz Pilot System Installation
- Configuring Template Wizard settings

4.4   Video Codecs

If you are previewing proxy versions of video from Viz One using Viz Engine (with Viz Engine installed on your control client machine) you must install video codecs. These are not part of Vizrt's standard installation.

Please check the section *Supported Codecs* in the Viz Artist User Guide for details.

> ❌ **IMPORTANT!** Due to licensing requirements, Vizrt does not provide the codecs required for local preview. Users must obtain and install their own codecs.

4.5   Crop Service Installation

Pilot Data Server must first be installed, see Pilot Data Server Installation, before Crop Service can be installed.

> ⚠ **Note:** Crop Service is a 64-bit application, so it must be installed on a 64-bit machine. To simplify configuration, it's recommended that Crop Service be installed on the same machine as Pilot Data Server.

4.5.1   Installing Crop Service

> ⚠ **Note:** Pilot Data Server and Crop Service must have the same version number (major.minor).

- Run the Crop Service installer: `VizrtPilotCropService-x64-x.y.msi`

4.6   Pilot Data Server Installation

Pilot Data Server gives you access to Viz Pilot's database and other services like Crop Service, Template Tagging, Update Service, Person Search, search on Viz One, Thumbnail Generator and Timeline Editor. It's also required when using Graphic Hub as the Viz Pilot database, or for integrating with Viz Story.

4.6.1   Installing Pilot Data Server

1. Click the Data Server installer file (*.msi) to start the *Data Server Setup Wizard*.
2. In the **Welcome** panel, click **Next**.
3. In the **Destination Folder** panel, define the location of the Data Server program files, and then click **Next**.

> ⚠ **Note:** The default location is % `ProgramFiles%\Vizrt\Pilot Data Server\`

4. In the **Setup Database Connection** panel, define the following settings:
   - **Connect string**: Connection string (`<host name>/<service name>`) to the Viz Pilot database.

     > ⚠ **Note:** When installing the Data Server on a new system, the default database configuration setting points to `localhost/vizrtdb` .

   - **Username:** Viz Pilot database username.
   - **Password:** Viz Pilot database password.

     > ⚠ **Note:** This will allow you to edit all database settings for all clients connected to the same database.

5. Click **Next**.
6. In the **Ready to install** panel, click **Install**.
7. In the **Completed Setup Wizard** panel, click **Finish**.

Silent Installation

Using silent installation allows the Data Server to be installed without using the installer interface.

In a command shell, run the following command:

```
msiexec /i VizrtPilotDataServer-x64-x.x.x.x-Release.msi /quiet
```

Thumbnail Generator

Thumbnail Generator can be installed as an optional component during the Data Server installation. The option for Thumbnail Generator installation is disabled by default. To install, select the Thumbnail Generator manually during the Data Server installation.

Silent Installation

Since the Thumbnail Generator will not be installed by default, INSTALLLEVEL =2 needs to be added to the command string for the Thumbnail Generator to be included during a silent Data Server installation:

```
msiexec /i VizrtPilotDataServer-x64-x.x.x.x-Release.msi INSTALLLEVEL=2 /quiet
```

To upgrade an existing Data Server to include the Thumbnail Generator without uninstalling it, run the following command:

```
msiexec /i VizrtPilotDataServer-x64-x.x.x.x-Release.msi
ADDLOCAL=ThumbnailGeneratorService /quiet
```

4.6.2 Notes

- If Crop Service Installation is required, it must be installed separately after the Data Server has been installed.
- If support for multiple Viz Pilot database schemas is required, *separate* Data Servers must be installed on *separate* machines.

4.6.3 Firewall Exceptions

In order for Data Server to be accessible from other machines, Windows firewall exceptions are required.

Data Server adds the following inbound rules to the Windows firewall during installation: allow connections on port 8177, 7373, and 9876. These are removed when Data Server is uninstalled.

4.7 Viz Pilot System Installation

This section covers:

- Viz Pilot Setup Wizard
- Installing Viz Pilot
- Installing Preview Server
- Installing Pilot DBA
- Viz Pilot Configuration

4.7.1 Viz Pilot Setup Wizard

The Viz Pilot Setup Wizard installs the necessary Viz Pilot components on your machine.

> ⚠️ **Note:** The installer migrates any configuration settings from older versions of Viz Pilot, such as Database information. This applies to the installer for Viz Pilot 7.1 and newer and only migrates from versions 6.0 and up. Settings from the most recent version of Viz Pilot will be migrated in the event that multiple versions are installed on your machine. For example, settings from Viz Pilot 7.0 will migrate but not from version 6.1, if both are installed.

4.7.2 Installing Viz Pilot

> ⚠️ **Note:** Microsoft .NET Framework 4.5.2 is required in order to run the Viz Pilot installer.

1. Click the Viz Pilot installer (*.msi) file to start the *Viz Pilot Setup Wizard*.
2. In the **Welcome** panel, click **Next**.
3. In the **Destination Folder** panel, define the location of the Viz Pilot program files, and then click **Next**.

> ⚠️ **Note:** The default location is `%ProgramFiles(x86)%\Vizrt\Viz Pilot x.y`.

4. In the **Product Features** panel, select one or more relevant Viz Pilot Components to install, and then click **Install**.
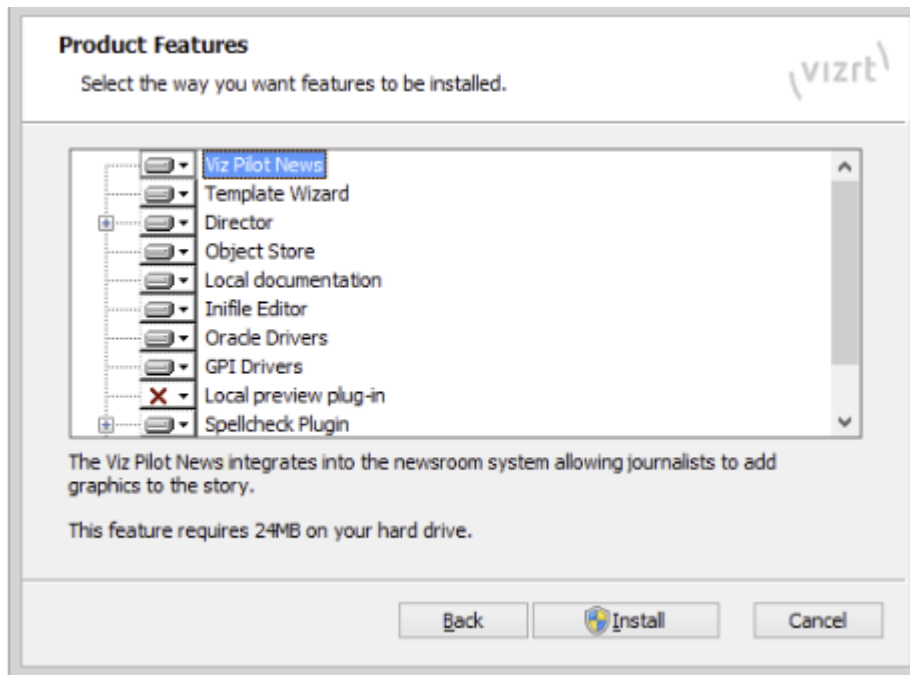
> ✅ **Tip:** Different components are needed for different setups. For example, on a Template Wizard work station or newsroom work station, only some Viz Pilot Components are needed. For more details, see Newsroom Integration Installation, Control Room Client Installation, and Template Design Client Installation.

5. Once Viz Pilot Setup Wizard has finished installing all relevant features, continue the installation process by configuring your setup in the Changing Settings.

4.7.3   Installing Preview Server

Preview Server is required in situations where the Viz Engine must provide frames for snapshot or thumbnail generation. Preview Server is used by Viz Pilot News to fetch previews of overlay graphics for the Timeline Editor.

See Also

- The installation section in the *Preview Server Administrator Guide*.

4.7.4   Installing Pilot DBA

Pilot Database Administrators (DBA) is used to install, maintain and upgrade the Viz Pilot database. There is no installer. The DBA can be downloaded from Vizrt's FTP as a ZIP archive, and is located in the same directory as the Viz Pilot installer. Pilot DBA and all script files are bundled into a ZIP file. Please unzip all files into one folder, for example: *C:\temp\PilotDBA*. Then:

1. Extract the ZIP archive to a preferred directory.
2. Create a desktop and/or start menu shortcut.

4.7.5   Viz Pilot Configuration

When all the relevant Viz Pilot Components have been installed using the Viz Pilot Setup Wizard, Changing Settings opens automatically.

In the Viz Pilot Configuration Tool, you can define the relevant settings for your Viz Pilot setup, for example:

- Configuring the Database Connection
- Configuring the Media Sequencer Connection
- Configuring Preview Settings
- Configuring Newsroom Settings
- Configuring Template Wizard Settings

4.8 Unattended Installation

Easily install or upgrade your system. Unattended installations are typically used to save time for large-scale roll-outs. This section covers the following topics:

- Configuration Files and Settings
- Configuring Viz Pilot in Non-GUI Mode
- Creating a Configuration Bundle
- Installing a Configuration Bundle
- Unattended Installation from the Command Line
- Remote Unattended Installation Using Active Directory
- MSI Properties
- Pilot Data Server Unattended Installation

4.8.1 Configuration Files and Settings

The Viz Pilot system is configured through a combination of local configuration settings on each workstation running the client software and the list of VCP parameters available in your Pilot Data Server web interface.

Local configuration is defined in two locations:

1. Some configuration files are stored in the `%PROGRAMDATA%\Vizrt\Viz Pilot x.y\Config` folder.
2. In addition, a number of *Registry keys* also contain relevant information:

- `HKEY_LOCAL_MACHINE\SOFTWARE\Wow6432Node\\[vizrt]\ActiveX x.y`

- `HKEY_LOCAL_MACHINE\SOFTWARE\Wow6432Node\\[vizrt]\viz|content pilot x.y`

- `HKEY_LOCAL_MACHINE\SOFTWARE\Wow6432Node\\[vizrt]\Preview Engine`

  To apply a previously prepared configuration to a Viz Pilot installation, a Configuration Bundle has been created (hereafter referred to as *bundle*). A bundle is a zip-file that contains a copy of all the configuration files and configuration values from the Registry, and some metadata (version information, etc.). The bundle can be read by the Changing Settings and applied to the current system.

4.8.2 Configuring Viz Pilot in Non-GUI Mode

The Changing Settings is primarily used to provide an interface (GUI) for user configuration of Viz Pilot Components. The tool opens automatically on regular installation, and can subsequently be opened manually. The Changing Settings also has a non-GUI mode of operation used to manage bundles.

Installing a bundle requires it to be run from a CMD shell with elevated privileges:

1. Click the **Windows** key, and then type *cmd*.
2. Right-click the entry `cmd.exe` and select **Run as administrator**.
   You should now have a command prompt that is elevated. This can be verified by looking for the *Administrator:* prefix in the window title.
3. Navigate to the installation folder where you installed Viz Pilot, and then to the sub-folder **Configuration**. If the default paths are used on a 64-bit system, this will be `%ProgramFiles(x86)%\Vizrt\Viz Pilot x.y\Configuration`.

4. You can then run `PilotConfiguration.exe --help` to see the help text for the program. The help text indicates which options require elevated privileges. An error message appears if you're running an option that requires elevation from a normal CMD shell.

All operations performed by the configuration program are logged in the file `%ProgramData%\Vizrt\Logs\Viz Pilot x.y\configuration.log` . By default, only important messages are shown on the console, but if you wish to see the same verbose output, append `--verbose` to any command line used.

> ✅ **Tip:** If you want to create a bundle, the bundle can be run from a normal CMD shell without selecting the *Run as administrator* option.

4.8.3   Creating a Configuration Bundle

You must first configure one workstation to the correct state. You may need to change configuration settings and test the changes a few times before the configuration is acceptable, before creating the bundle.

Creating the Bundle

1. Run `PilotConfiguration.exe --createbundle=%USERPROFILE%\Desktop\config.zip` . This will create a zip-file called *config.zip* on the current user's Windows desktop. Note that if the file already exists, an error message will appear and the Changing Settings will close. If that occurs, either delete the existing file or modify the filename before running the command again.

An ordinary zip file is created. You can unzip the file to see or alter the configuration. For example, if you do not want to distribute the keyboard settings, you can delete the file from the bundle.

> ⚠️ **Note:** When editing bundles manually, be precise and always test your changes thoroughly before rolling them out.

4.8.4   Installing a Configuration Bundle

To apply the settings in an existing bundle to your current workstation, use the Changing Settings's option *installbundle* as follows:

1. `PilotConfiguration.exe --installbundle=%USERPROFILE%\Desktop\config.zip` . This overwrites any local configuration with the configuration from the bundle.

> ✅ **Tip:** For detailed information about what has been changed, use the `--verbose` option.

> ⚠️ **Note:** Applying settings from a bundle removes local settings only - not those in the bundle. For example, if the local system contains a `keyboard.cfg` file, but the bundle does not, the local configuration file will be deleted. The same applies registry values under the keys mentioned above.

4.8.5   Unattended Installation from the Command Line

Since the Viz Pilot system is distributed as a Microsoft Installer package, the standard MSI tool has been used to facilitate an unattended installation. The command `msiexec.exe` can be used for this. Although a full tutorial on how to use `msiexec` is not provided below, options specific to the Viz Pilot system's installer are covered.

To run an unattended installation using default options, showing only a progress-bar, run the following command:

```
msiexec /qb /i VizPilot-x.y.z.z.msi
```

This runs the installer in Basic UI mode, where the overall progress bar is displayed and the Changing Settings is not displayed. The installer will be installed in the default folder. Of course, since installing with default settings is not always the desired behavior, hooks can be used to select the install location, set the list of components to be installed, and supply the installer with a configuration bundle to install.

This section contains the following topics:

- Installation Location
- Feature Selection
- Configuration Bundles
- Icons on the Windows Desktop
- Customized Installations

Installation Location

The directory where the bulk of the software will be installed is controlled through an MSI property called **INSTALLDIR**. By default, this is either `%PROGRAMFILES%\Vizrt\Viz Pilot x.y` if no earlier version of Viz Pilot has been installed, or the installation folder of the earlier Viz Pilot version. Some files will be installed outside of this folder, either in the Windows system folder or the `%PROGRAMFILES%\Vizrt\Common` folder. These files cannot be moved by changing the **INSTALLDIR** property.

To override the **INSTALLDIR** property using `msiexec` you append `INSTALLDIR="C:\the\correct\folder"` to the `msiexec` command line. For example:

```
msiexec /qb /i VizPilot-x.y.x.x.msi INSTALLDIR="D:\Software\Viz Pilot x.y"
```

If an earlier Viz Pilot version is already installed, the installation will be moved to the new folder, in the same way as is done when selecting a new folder when running the installer interactively.

Feature Selection

Installing all the features of the Viz Pilot system's installer is not required when running unattended. Select which features are to be installed by listing the features by name. Since feature names are not easily extracted from the MSI, a list has been included below. In a similar manner to setting the install location, supply a comma-separated list of the required features in an MSI property named **ADDLOCAL** on the command line:

```
msiexec /qb /i VizPilot-x.y.x.x.msi ADDLOCAL="Director,GPIDrivers"
```

This will install Viz Pilot's Director component and the drivers for GPI integration only.

Viz Pilot Components in Unattended Installations

Feature ID	What will be installed
Common	Common files for all the features. Note: This is always automatically installed and need not be specified.
Documentation	The PDF and HTML documentation
Director	Director control room client
TimecodeMonitor	Timecode Monitor
TrioPlugin	Viz Trio Data Resource Plugin
VizPilotNews	Viz Pilot News
TemplateWizard	Template Wizard
ObjectStore	Object Store
OracleDrivers	Oracle Drivers
GPIDrivers	GPI Drivers for Director
LocalPreviewPlugin	Local Preview Plugin for Director and Viz Pilot News
SpellcheckPlugin	Spell check plugin
SpellcheckPluginDictionaries	Default dictionaries for the Spell check plugin
QuickCG	Quick CG

Configuration Bundles

It's also possible to specify a configuration bundle during an unattended install. This is also done through MSI properties, using a property named **CONFIGBUNDLE**.

```
msiexec /qb /i VizPilot-x.y.x.x.msi CONFIGBUNDLE="%USERPROFILE%\Desktop\config.zip"
```

This will install Viz Pilot in the default location, with the default set of features, and the configuration from the `config.zip` bundle located on the current user's desktop.

Note that when specifying remote paths in the **CONFIGBUNDLE** property, network drive mappings on Windows are on a per-user basis. Depending on your specific setup, a drive named `Z:` might not exist for the user running the installation. To avoid issues, specify remote resources with a UNC path (for example `\dist-server\vcp\config.zip`) and make sure the network share is readable by the machine account.

Icons on the Windows Desktop

To place icons on the current user's desktop during installation, use the MSI property **INSTALLDESKTOPSHORTCUTS**. For example:

```
msiexec.exe /qb /i VizPilot-x.y.x.x.msi INSTALLDESKTOPSHORTCUTS=1
```

Customized Installations

All these options can be specified together to customize the installation to your specifications. For example:

```
msiexec /qb /i VizPilot-x.y.x.x.msi CONFIGBUNDLE="%USERPROFILE%\Desktop\config.zip"
  ADDLOCAL="VizPilotNews" INSTALLDIR="D:\Software\Viz Pilot x.y"
```

4.8.6   Remote Unattended Installation Using Active Directory

There are multiple ways to perform remote unattended installations of software packages in Windows networks. The distribution of software through Active Directory and Group Policy Objects has been chosen as an example. The techniques used here can be adjusted to match your specific setup.

We have used Windows Server 2012 R2 in this example. The exact names of the tools used may differ if you're using a different version of Windows Server.

This section contains the following topics:

- Background
- Creating a Transform
- Creating a GPO with an Applied Transform for Distributing Viz Pilot
- Troubleshooting

Background

In a Windows domain, settings for different client computers are distributed in *Group Policy Objects (GPOs)*. These are self-contained units of configuration. One of the things they can specify is the installation of packages in MSI format, which is what we will use.

One limitation here is that you cannot define arbitrary MSI properties like you can on the command line examples using `msiexec.exe`, so you need to use an alternative configuration method. MSI provides a mechanism called a *transform*. A transform is a collection of changes applied to an installation. By applying a transform to a base installation package, the installer can add or replace data in the installation database. The installer can only apply a transform during an installation. The tools used to create GPOs allow the specification of a transform to apply to a package, providing an opportunity to apply customizations.

Creating a Transform

One way to create a transform using a freely available tool, *Orca*, is shown below:

1. First, start Orca.
2. In the **File** menu, click **Open**.
3. Navigate to the Viz Pilot installer, and open it.
4. Select **Transform > New Transform**.
   The title of the Orca window should now say *(transformed by Untitled)*.
5. Locate the **Property** table in the listing on the left-hand side, and select it.
6. Right-click the property list and select **Add Row**.
7. Enter a property name of `INSTALLDIR`.
8. In the next window, enter the value of `E:\Vizrt\Viz Pilot x.y`.
   This sets the installation location.
9. Right-click again and add a row with the name `ADDLOCAL` and value `VizPilotNews,ObjectStore`.
   This is a comma-separated list of feature IDs that will be installed. In this example, we install Viz Pilot News and the Object Store client only (nothing else is installed, including bundled Oracle Instant Client libraries.)
10. Repeat this procedure (if required) for the `CONFIGBUNDLE` property.
11. When you are satisfied with the new values in the property table, select **Transform > Generate Transform**. Save the transform file to the preferred location.

Creating a GPO with an Applied Transform for Distributing Viz Pilot

We will now show you how to use the transform, a configuration bundle and Viz Pilot MSI installer to create a GPO. Since the example uses Windows Server 2012 R2, the exact steps might vary according to the specific version of Windows Server being used.

First, prepare a network share to host the MSI, transform and configuration bundle. This network share must be readable by the target client's computer account, since there will be no interactive user logged in to lend their access credentials to the installer process when the install is being run unattended. Although this can be done by setting the share as read-only to *Everyone*, the details and best way of doing this will differ according to the network.

In this example, we will use the UNC path `\server\software` to represent this share. Put all three files on this share. The three paths are:

- `\server\software\VizPilot-x.y.x.x.msi`
- `\server\software\transform.mst`
- `\server\software\config.zip`

You might end up having multiple `transform.mst` and `config.zip` pairs to support different sets of configuration in your setup.

First, create a GPO representing the software installation.

1. Start the **Group Policy Management** tool.
2. Place the GPO in the **Group Policy Objects** container.
   We are using the domain structure, so this container is preferred here, but there might be a better location for it in your domain.
3. Right-click the **Group Policy Objects** container, and select **New**.
4. Give it a descriptive name, for example `Viz Pilot installation`.

> ⚠ **Note:** Make sure the **Source Starter GPO** is set to *(none)*.

5. Click **OK**.
6. Right-click the GPO we just created in the **Group Policy Objects** container, and click **Edit**.
7. In the **Group Policy Management Editor** window that opens, navigate to **Computer Configuration > Policies > Software Settings**.
8. Right-click it and select **New > Package**.
9. Using the file selector that appeared, navigate to the installer file.

> ⚠ **Note:** It is important that you access the GPO through the network share, even though it might also be visible through the local disk.

10. In our case we navigate to and select `\server\software\VizPilot-x.y.x.x.msi`.
11. In the box that appears, click **Advanced**.
    The properties windows for this package will now be shown.
12. Go to the **Modifications** tab and click **Add**.
13. Navigate to and select the transform file.

> ⚠ **Note:** Again, make sure to use the network share.

14. Click **OK**.
15. Close the **Group Policy Management Editor** window.
    You now have a GPO that is configured to install Viz Pilot, but it needs to be assigned to a set of clients to take effect. In our case, all our relevant clients are in the **Clients** container on the top level of the domain, and we will link the GPO to this container.
16. In the **Group Policy Management** window, right-click the container in the list on the left hand side, and then click **Link an Existing GPO**.
17. Select the GPO you just created in the list, and click **OK**.
    The clients should now install the software on the next reboot.

Troubleshooting

Using this method of installing software has one major drawback: there is no reporting back to a central location about installation issues or configuration errors. For installation issues related to the MSI installer itself, you will need to enable general logging of all MSI installations. If the installer finishes, but something is wrong in the configuration, the Viz Pilot Configuration program logs to the file `%PROGRAMDATA%\Vizrt\Logs\Viz Pilot x.y\configuration.log`.

4.8.7   MSI Properties

- **INSTALLFOLDER:** The location where Viz Pilot will be installed. Note that various support files are not affected by this, for example files that must be deployed to the Windows system folder or the files in `%PROGRAMFILES%\vizrt\Common` .
- **CONFIGBUNDLE:** The path to a configuration bundle that must be applied to the system during installation.
- **ADDLOCAL:** The list of feature-names to install.
- **INSTALLDESKTOPSHORTCUTS:** If this is set to any value, the installer will place shortcuts to installed programs on the current user's desktop.

4.8.8   Pilot Data Server Unattended Installation

Pilot Data Server configuration values can be provided on the command line when performing an installation, so that they do not need to be entered manually in the installer GUI. Specifying the values on the command line is primarily done during a silent/unattended installation, using the Windows Installer parameters *quiet* or *passive*. See Microsoft's Windows Installer documentation for more options.

See also Pilot Data Server Installation.

Pilot Data Server with Graphic Hub Database

Use the following parameters to install Pilot Data Server with a **Graphic Hub** database:

- `CONNECTION_METHOD="GH"` : Indicates that the Pilot database is running on a Graphic Hub
- `GH_HOST` : Hostname of the Graphic Hub REST server
- `GH_PORT` : Graphic Hub REST port
- `GH_USER_ID` : Username for accessing the Graphic Hub REST server
  For example:

```
msiexec /i <path/to/msi> CONNECTION_METHOD="GH" GH_HOST="localhost"
GH_PORT="19398" GH_USER_ID="Guest" /quiet
```

Pilot Data Server with Oracle Database

Use the following parameters to install Pilot Data Server with an **Oracle** database:

- `CONNECTION_METHOD="Oracle"` : Indicates that the Pilot database is running on Oracle
- `ORACLE_USERNAME` : Oracle database username
- `ORACLE_PASSWORD` : Oracle database password
- `ORACLE_DATA_SOURCE` : Oracle data source containing Oracle server host and the database (or instance) name
  For example:

```
msiexec /i <path/to/msi> CONNECTION_METHOD="Oracle" ORACLE_USERNAME="pilot700"
ORACLE_PASSWORD="pilot700" ORACLE_DATA_SOURCE="bdvcpdb/vizrtdb" /quiet
```

5   Configuration

In order to have a working Viz Pilot system, most of the applications must be configured to use a Media Sequencer, a database and a Viz Engine for playout and preview.

This section contains the following topics:

- Changing Settings
- Viz Engine Preview Configuration
- Initialization and Configuration Files
- Database Configuration
- Oracle Database
- Graphic Hub Database
- Database Parameters
- Registry Settings

5.1   See Also:

- Director Configuration
- Storage and Backup

5.2   Changing Settings

To configure Viz Pilot and change settings, access the configuration tool from the Windows Start Menu (**Viz Pilot Configuration**) or by clicking the desktop icon:

> ✅ **Tip:** The Viz Pilot Configuration window opens automatically when the Viz Pilot Setup Wizard has finished installing all relevant Viz Pilot Components. It is also possible to open Viz Pilot Configuration in non-GUI Mode.

This window shows any Viz Pilot Components that are installed. For example, the **Wizard** tab shows that Template Wizard is installed.
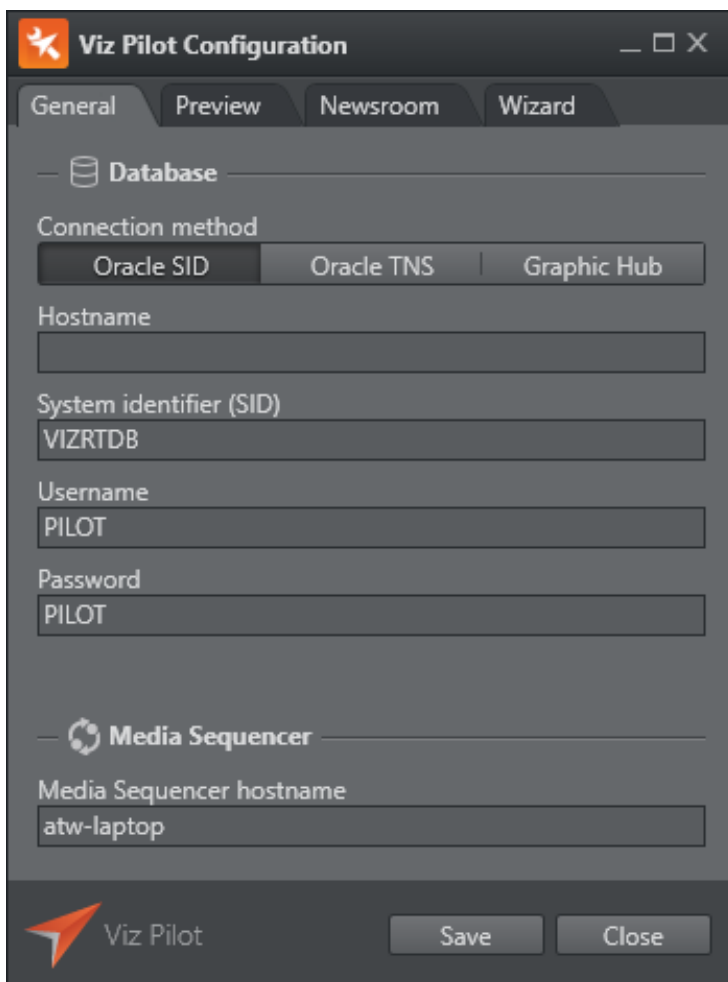
This section describes configuration options:

- General
- Preview

- Newsroom
- Wizard
- Font Settings

5.2.1   General

Click the **General** tab in the Viz Pilot Configuration tool to configure settings related to the Database and Media Sequencer.

Configuring the Database Connection

Database settings changes must be done in this tab and cannot be changed from inside Director. Since the database is key to any Viz Pilot system setup, it is important to do this correctly. If you do not have the correct information at hand, please contact your database administrator before making any modifications to the database settings. All database connections for Viz Pilot 5.x and later must use UTF-8 character encoding.

Enter the details for the type of connection you require.

Oracle Database

- **Oracle SID:** This option allows for a direct connection to an Oracle database, without a configured `tnsnames.ora` file on the system. Enter a valid **Hostname** and **System identifier (SID)** (default is *VIZRTDB)*, database **Username** (default is *PILOT*) and **Password** (default is *PILOT*).
- **Oracle TNS:** This option allows for a connection to an Oracle database using a TNS alias only, if there is a correctly configured `tnsnames.ora` file on the system. Enter a valid **TNS alias** and directory **TNS path** for the Oracle configuration file `tnsnames.ora` . For more details, see Database TNS Alias. Enter the database **Username** (default is *PILOT*) and **Password** (default is *PILOT*).

> ✅ **Tip:** In order to use Viz Pilot's database locally, configure an Oracle Instant Client during installation.

Graphic Hub Database

- The **Graphic Hub** type allows for a connection to a Graphic Hub database. Enter the **Graphic Hub REST hostname** and **Port** (default is *19398*), and the **Username** for the user that has been configured in the Graphic Hub for Viz Pilot connections (default is *Guest*).
- Check **Use HTTPS** to make Template Wizard and Director communicate over HTTPS with TLS protocol version 1.2 to the GH REST service.

> ⚠️ **Note:** In Viz Pilot 7.0, the username does not require a password. The user and its group must be active and the user needs at least user write access. For information on how to manage users and security, see the Graphic Hub User Guide.

Configuring the Media Sequencer Connection
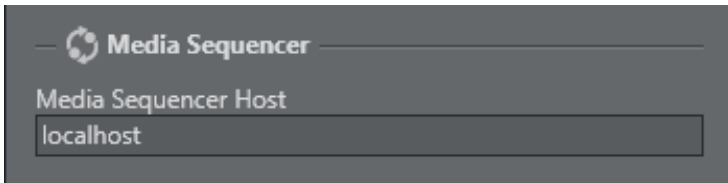
In the **Media Sequencer** section, enter the **Media Sequencer Host** name (or IP address).

> ⚠️ **Note:** To change the database connection for the Media Sequencer, in Director, go to Tools > Media Sequencer Configuration > Database Tab.

5.2.2   Preview

Click the **Preview** tab in the Viz Pilot Configuration tool to configure the preview of template graphics. The settings in this panel define the preview in Director and Viz Pilot News. Database settings apply to all clients that do not have preview configured locally. These preview settings do not affect the preview inside the Timeline Editor.

Configuring Preview Settings
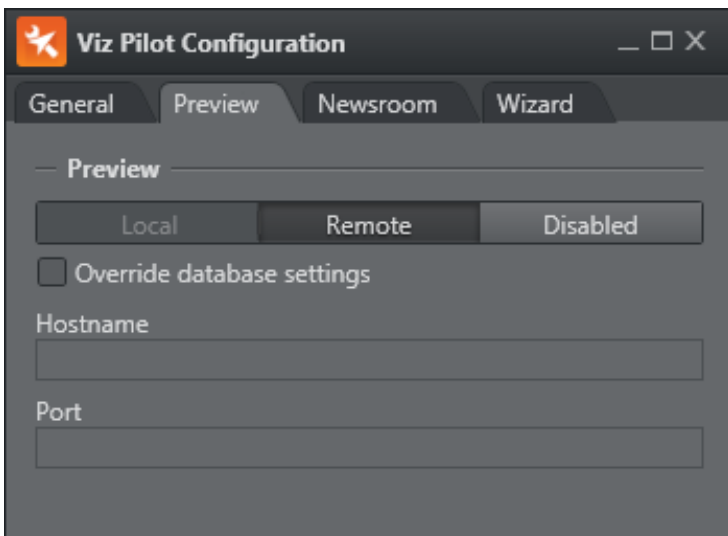
- **Local:** This option uses a locally installed Viz Engine (requires a license) and is only available if the *Local preview plugin* has been installed as one of the Viz Pilot Components .
- **Remote:** This option uses a remote Viz Engine (recommended). A **Remote** preview *without* the **Override database settings** means that the remote host and port is taken from Viz Pilot Parameters `ax_preview_host` and `ax_preview_port` .
- **Disabled:** This option disables the preview functionality altogether.
- *Optional:* Select the **Override database settings** box. This option will override the Viz Engine settings specified in the database. Enter the valid Viz Engine preview **Hostname** (or IP address) and the **Port** number. In a remote preview setup, port *50008* is typically used.

> ✅ **Tip:** Multiple renderers can be defined as a comma-separated string. For example *<host>, <host>, ...* or *<host>:<port>,<host>:<port>, ...* . If hosts are defined without a trailing port number, it is recommended to set the default port in the Port field. If no port number is set, it will default to 6100.

> ⚠️ **Note:** Configuring the preview settings will add the settings to your local machine's registry. If you have *multiple newsroom clients*, all clients can use the preview host settings by adding them to your Database Parameters.

5.2.3   Newsroom

Click the **Newsroom** tab to configure newsroom settings.

Configuring Newsroom Settings
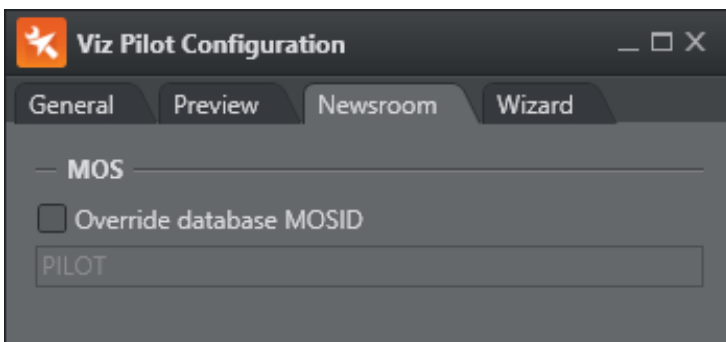
- **Override database MOSID:** Selecting this option will override the database settings configured for MOSID. Enter a valid MOS ID.

> ✅ **Tip:** The MOS ID can be obtained from your Newsroom Computer System. The default MOS ID is *PILOT*. Note that most newsroom systems generate case sensitive MOS IDs.

> ⚠️ **Note:** The installer in Viz Pilot 5.7 and earlier had the option to set some Advanced iNEWS settings during installation, in order to configure VizDM and iNews ControlAir Device Manager. Although these are no longer included in the installer, the values (insertBCSTag, BCSDeviceName, UseVCPDM, BCSTemplateName) can still be accessed as Registry Settings.

> ⚠️ **Note:** For manual configuration of the MOS ID in Director, see the ContentPilot.ini's MOS section. Viz Pilot News configures the MOS ID using the following registry value:
> ```
> HKEY_LOCAL_MACHINE\SOFTWARE\Wow6432Node[vizrt]\ActiveX
> x.y\TemplateFiller\mosID
> ```

5.2.4   Wizard

Click the **Wizard** tab to define the list of available Viz Engines in the Template Wizard.

> ⚠️ **Note:** This tab is only available if Template Wizard has been installed as one of the Viz Pilot Components.

Configuring Template Wizard Settings

> ⚠️ **Note:** Template Wizard requires that a Media Sequencer is running on the same machine.
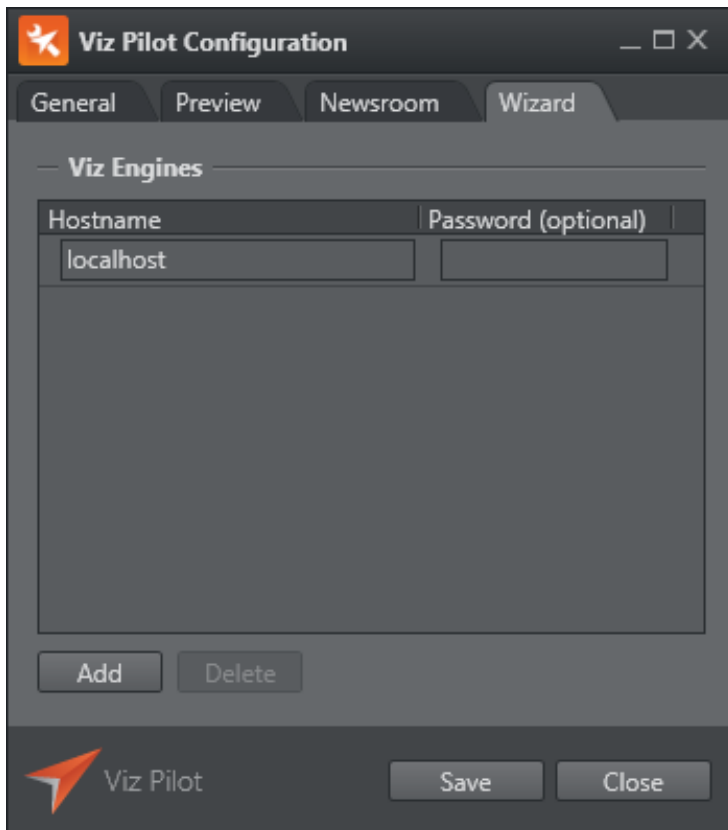
Define the **Hostname** and **Password (optional)** of your Viz Engines.

In order to be able to work with creating templates for scenes, Template Wizard connects to a Viz Engine, which in turn connects to a Graphic Hub. The default hostname is *localhost*. There is no need to modify this hostname if you have a Viz Engine installed locally. If you wish to connect to remote Viz Engines, simply add the hostnames (or IP addresses).

The template and data list font settings for Director are set in the initialization (INI) file: ContentPilot.ini; for Viz Pilot News this is done in the registry.

- **font:** Sets the font for Director. This is set to MS Sans Serif by default but can be changed.
- **ListDisplayFont:** Sets the font for Viz Pilot News. This string value is not set by default, and can therefore be used to override the operating system defaults.

5.2.5   Font Settings

Set the Display Font for Director

1. Open the **ContentPilot.ini** file.
2. Select the **LANGUAGE** section and set a new Unicode **font** parameter.
3. Save the file, and restart Director for the changes to take effect.

Set the Display Font for Viz Pilot News

1. Open Registry.
2. Search for the key:
   - Windows 64-bit: `HKEY_LOCAL_MACHINE\SOFTWARE\Wow6432Node\ [vizrt]`
   `\ActiveX\TemplateFiller`

   Add the string value **ListDisplayFont** with the desired font name as parameter.

5.3   Viz Engine Preview Configuration

Local preview uses a locally installed Viz Engine and requires installation of the *Local Viz Engine Preview* plugin.

Remote preview uses a remote Viz Engine which can provide either a real-time or a snapshot preview.

5.3.1   Real-Time Preview

The real-time preview option uses a Viz Engine with a reference monitor for animated and snapshot still previews. This is the recommended solution for graphics designers and control room operators. Real-time preview works best using an extra Viz Engine as this allows you to pass through your video input. Rather than using an extra Viz Engine, two Viz Engine instances can be run on one physical machine (provided you have two graphics cards in a dual channel setup). Alternatively, you can configure the Viz Engine used for program output to feed still previews using port 50010; this can be configured from Profile Configuration in Director.

> ⚠ **Note:** In control rooms, it is recommended to use a stand-alone Viz Engine with a reference monitor for full graphics and video preview.

5.3.2   Snapshot Preview

The remote snapshot preview option uses a shared Viz Engine with the option of showing stills, animations and preview points. This is the recommended solution for multiple simultaneous users (for example for thumbnail generation, newsroom and non-linear editing users).

Remote Snapshot Preview allows you to see a frame by frame preview of either selected frames or frames of preview points added to the scene by the graphics designer. This option does not support preview of your video input.
It is recommended to add the **pilot1** tag on the scene's default (main) director in order to set a specific preview or generation point. This also applies transition logic scenes that must have the tag set for the foreground scenes. The **pilot1** tag is case sensitive and MUST be lowercase.

You may use Director's built-in Remote Preview (**CTRL+P**) window for a Remote Snapshot Preview. This can be configured in the Director Preferences window and must not be confused with Viz Pilot's Preview plugin which requires a local Viz Engine.

This section covers the following topics:

- Real-Time Preview
- Snapshot Preview
- Configuring Local Preview
- Configuring Remote Preview for Newsroom
- Configuring Preview in Director

5.3.3   Configuring Local Preview

Local preview requires installation of the Local Viz Engine Preview plugin. The local Viz Engine used for preview rendering should preferably have the same configuration and hardware setup as the Viz Engine used for program rendering. Local preview is not a recommended setup if the machine does not meet the hardware requirements.

Local preview is not enabled by default, and must be enabled/disabled in the Changing Settings.

If you are using Viz Engine to preview video clips from Viz One, video codecs must be installed on Viz Engine, see Video Codecs. Note that such codecs installation only applies when Viz Engine is used for previewing *low-resolution proxy versions* of video clips stored on Viz One. Playout of high-resolution versions does not require additional codecs installation.

Ports for Local Preview

For local preview, it is recommended to configure a Viz Engine to use default port 6100. If local preview is required, it is recommended to use the Viz Preview License server in order to minimize the need for hardware license dongles.

If the port number is changed, Viz Engine must change its communication port as well. Use **Viz Config** to set the new communication port number.

Configuring Local Preview

1. First, see how To add a Viz Engine in Director, and then perform the following steps:
   a. **Add** the newly created Viz Engine **to your channel** and check the Viz Preview check-box. Use **localhost** as your hostname.
   b. Click **Close**.

Automatically Log In to Graphic Hub

> ❌ **IMPORTANT!** Users MUST configure Viz Engine to automatically log in to Graphic Hub, since Director and Template Wizard are not able to log in automatically.

Prerequisite: For automatic login to work, an initial login to a Graphic Hub via Viz Engine must be performed before local preview will work. During this process, the hostname, Graphic Hub and User must be entered. For more information see the *Viz Artist User Guide* .

1. Start **Viz Config**.
2. In the **Database** section enable **Auto Login**.
3. Click **Save** and close Viz Config.

5.3.4   Configuring Remote Preview for Newsroom

Ports for Remote Preview

The recommended option is to configure a second Viz Engine to use default port 6100 (enabling real-time preview on a reference monitor).

For remote preview where you want to support *multiple users,* a remote Viz Engine can be configured to use preview port 50008 (enabling a client-side still/snapshot preview).

For remote preview where you need to use the *same* Viz Engine for both program and preview rendering, use port number 50010 (enabling a still/snapshot preview on a reference monitor).

By default, Viz Engine uses port number 6100 for both program and preview rendering. If a port number is changed, Viz Engine must change its communication port as well. Use **Viz Config** to set the new communication port number. Note that this is not required for the multiplexing ports (50008 or 50010).

> ⚠️ **Note:** Do not use a program renderer for Remote Snapshot Preview using port 50008.

Configuring Remote Preview in Viz Pilot News

Viz Pilot News uses Remote Preview by default, enabled automatically on installation.

If the Local Preview option has been used before, ensure that it is disabled in the Changing Settings, otherwise local preview will start instead of remote preview.

1. Go to the Pilot Data Server Web Interface. see the Pilot Data Server documentation.
2. Click the **Settings** tab.
3. Click the **VCP Parameters** tab.
4. Add or edit the `ax_preview_host` and `ax_preview_port` parameters:

- ax_preview_host: <hostname>
- ax_preview_port: 50008

> ⚠️ **Note:** Two or more hosts can be specified in CSV format, typically using a comma as a separator. Example: *ax_preview_host = myengine1,myengine2*. This feature is for basic load balancing only and does not add redundancy. All hosts must serve the same content.

Viz Engine Configuration Recommendations

The following are recommended configuration settings for the Viz Engine that is used as a Remote Preview for Viz Pilot News:

- Setup a Viz Engine VGA version.
- Use all the default configuration.
- Under Memory Management in Viz Config, set the following:
    - Free Image Data (settings depends on the graphics Card) can stay on = No.
    - Free Images = On.
    - Free Fonts = On.
    - Free Memory Threshold (MB) = 300.

Minor Memory and Speed Gains are Achieved by:

- Turning off audio (Audio Settings > Audio Active > Off).
- Deactivating Virtual Studio (Camera > Virtual Studio > Deactive).
- Setting the Output Format lower than house standard (for example from 1080i to PAL 16:9).
- Disabling plugins that are not in use.

5.3.5 Configuring Preview in Director

Configuring Remote Preview in Director

Director uses Remote Preview by default, enabled automatically on installation.

If the Local Preview option has been used before, ensure that it is disabled in the Changing Settings, otherwise, local preview starts instead of remote preview.

1. First, see how To add a Viz Engine, and then perform the following steps:
   a. **Add** the newly created Viz Engine **to your channel** and check the Viz Preview check-box.
   b. Click **Close**.

Configuring Remote Still Preview in Director

> ❌ **Warning:** This setting requires sufficient ring buffer on the program renderer to prevent the rendered still preview from causing the scene on air from dropping frames. This setting is therefore deprecated.

Still Preview allows you to use the program output to get a still preview, typically on a reference monitor. To achieve this, Media Sequencer creates a copy of the scene being read and sends commands to your program output renderer asking for a snapshot of the scene while the current scene on air is being rendered.
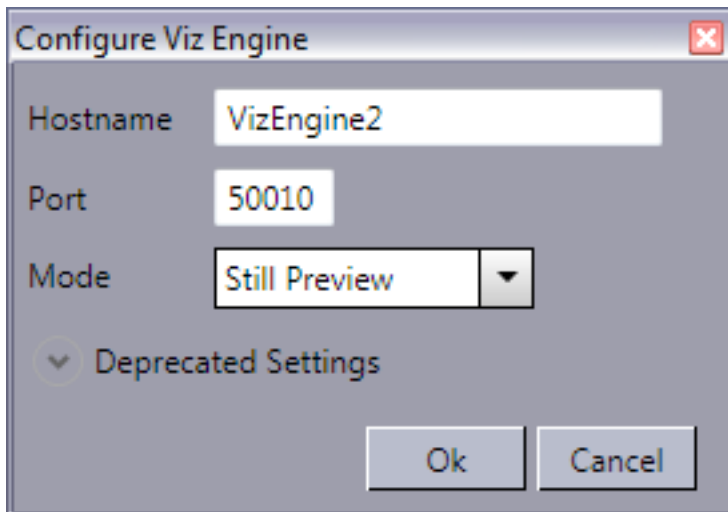
1. Create two channels, one for program and one for preview.
2. Configure the same render engine twice, as shown in how To add a Viz Engine.
3. For the second render engine set **Port** to **50010** and **Mode** to **Still Preview**.
4. Click **Ok**.
5. Add the program renderer to the program channel and the still preview renderer to the preview channel.

Configuring Remote Snapshot Preview in Director

1. Start Director.
2. Select Director Preferences from the **Options** menu.
3. Select the **Remote Preview** option in the Director Preferences window.
4. Enable the **Use this server** option to override the database settings:
   - Host: <hostname>
   - Port: 50008
5. *Optional*: Enable the **Always show "Animate" button** option, and set a preferred poll interval (seconds) for how often images should be retrieved from Viz.
6. *Optional*: Configure size and position of the remote preview window.

⚠️ **Note:** If the Viz Engine Preview plugin is enabled, local preview will start instead of remote snapshot preview. The plugin can be disabled in the Changing Settings.

5.4   Initialization And Configuration Files

The initialization files in the table below are used with the Viz Pilot system. Modified files are stored in *%PROGRAMDATA%\Vizrt\Viz Pilot x.y\Config*.

File	Description
ContentPilot.ini	Most Viz Pilot components use this *.ini* file. The default file is stored in the program folder during installation as `ContentPilot.ini`. See ContentPilot.ini.
objectstore.ini	Object Store may use its own ini file for reading and connecting to the Viz Pilot database. See Object Store Configuration.

In most cases, you can use the Changing Settings to make changes to the INI files. If required, they can be opened with a regular text editor, however, this is strongly discouraged.

5.4.1   ContentPilot.ini

ContentPilot.ini contains many settings, organized into the following sections:

- Button Box
- Database
- Configuration Files
- GPI
- Initialization and Configuration Files
- Viz IO

> ❌ **IMPORTANT!** Do not modify any settings that are not described in this document. Do not modify any settings unless required.

> ⚠️ **Note:** Deprecated settings are not described here.

Button Box

Viz Pilot button box settings:

- **enabled:** Set Y or N to enable or disable button box.

Database

- **login:** User ID for the database connection.
- **pwd:** Password for the database connection.
- **utf8:** Sets whether or not the program and database should use UTF8 font encoding. This value should not be modified.

- **nls_lang:** Sets the language to use. If UTF-8 is used for the database, the NLS_LANG environment setting must be here. This value is required and should not be modified.
- **name:** This entry must contain the connection string for the database in the form: "hostname/instance name". If a database client (such as Oracle Client) is used, enter the TNS name.

General

- **diff_tool:** The path to a text compare tool executable can be entered here. This will make it possible to compare script files when conflicts between local scripts and repository scripts appear.
- **imageheight:** Sets the default thumbnail image height generated by TnG.

GPI

This section contains GPI triggering settings for Director.

- **enabled:** Enable GPI triggering? Y or N.
- **poll:** Number of milliseconds between each poll of the GPI port.
- **delay_interval:** Number of milliseconds before repeating a GPI trigger.
- **use_lpt:** Use the LPT port for GPI triggering? Y or N.
- **send_to_cockpit:** Send the trigger to the cockpit? Y or N. If set, Viz Pilot will send a recall message to the Viz IO when a GPI is received.
- **allow_combination:** When sending a trigger to the cockpit, the allow_combination specifies if, for instance, GPI 3, should be sent as GPI 1, and GPI 2.

Import Sequence

Settings for enabling import of playlists from file (advanced).

- **enabled:** If set to Y, menu Playlists will be enabled.
- **default_file:** Sets the default file to import.

Language

- **font:** Sets the application's template and data list font. By default this setting will be set using the operating system font. However, in many cases other fonts, such as Arial Unicode MS, are used for language localization purposes (for example Arabic, Hebrew or Cyrillic).

⚠ **Note:** The font setting does not affect Viz Pilot News, where font settings are configured in the registry.

Parameters

- **datapool_visible:** Enable or disable the Datapool component in Template Wizard.
- **basic_properties:** Comma delimited list of additional properties to show in the basic properties tab in Template Wizard.
- **plugin_folder:** Folder where Template Wizard will look for *VcpExp.bpl*.

Path

This section specifies a set of file and folder paths required by the Viz Pilot system.

- **picture:** Defines the picture archive path.
- **production:** The directory where the images for today's production are placed - excluding the date.
- **autosave:** The directory where automatically saved template records are stored.
- **tmpbmp:** The directory where temporary bitmaps are stored.
- **scene_node:** Specify a node that is automatically selected when viewing the scene tree from Viz Engine in Viz Pilot.
- **image_node:** Specify an image that is automatically selected when viewing images from Viz Engine in Viz Pilot.
- **object_node:** Specify an object that is automatically selected when viewing objects from Viz Engine in Viz Pilot.
- **sounds:** The directory where WAV files are initially looked for.
- **shared:** The directory where the shared image disk is mounted on the Unix system. Used for specifying the Unix path to the shared image server. For instance, on windows the drive is often mapped as K:\ but on a Unix system it would be \shared\images.
- **fileopen:** Sets the default folder when opening image files.

Pilot

These settings define the general behavior of the Viz Pilot system.

- **restrict_characters:** Restrict characters when saving data.
- **legal_characters:** The characters that will be permitted in the names of saved data if "restrict_characters=Y". If this value is empty, there is no restriction on the data name regardless of the "restrict_characters" setting.
- **liveupdate_timer:** The interval in milliseconds between each time a template's Live Update event-handler is called. (If the template has Script Editor.)
- **dateformat:** The date format for saved data.
- **unique_names:** Whether or not it is permitted to save several data elements with the same name.
- **autostart_preview:** When set to Y, the Viz Pilot system will automatically start Viz [preview engine] if it is installed.
- **plugins:** The directory where the plugins for the Viz Pilot system are located. Normally, this value is empty, otherwise it should be the path to the directory.
- **UTF8:** Set to Y for UTF8 support in the Viz Pilot system.
- **logcount:** Number of rolling log files (default 0). Affects both the Viz Pilot system's log and macro log files.
- **logsize:** Maximum size of the log file in bytes (default 512000). Affects both the Viz Pilot system's log and macro log files.
- **logfile:** Override the Viz Pilot system log file name (default `vcp_yyyymmddThhmmss.log`).
- **logdir:** Override the directory the Viz Pilot system log files are put in. Default locations are:
    - `%PROGRAMDATA%\Vizrt\Logs\Viz Pilot x.y`
- **macrologfile:** Override the macro log file name.
- **macrologdir:** Override the directory where macro log files are located.

Sequencer

- **hostname:** Sets the hostname of the Media Sequencer when specified during installation. If no host is specified it is set to *localhost*.
- **locktoprofile:** When set to a specific profile, users will not be allowed to select other profiles or open the profile configuration in Director.

SGI

This section can be used to add new Viz Engines that can be used by Template Wizard.

- **machine1:** The host name of the Viz Engine to connect to. More machines can be added to the list: machine2, pwd2 etc.
- **pwd1:** On air password for the Viz Engine. Note that this is not supported on Viz Engine 3.5 or later.

Template_Wizard

- **linelayout_visible:** Enables the Prefix and Customize options in the Wizard tool that sets the number of prefixes to be used when generating a template with a table style layout.
- **datapool_visible:** Enables or disables the use of the datapool component in Template Wizard.
- **basic_properties:** Comma delimited list of additional properties to show in the basic properties tab.
- **use_database:** If disabled, the Template Wizard will operate without a database connection. This can be useful when creating file-based templates with Template Wizard. For example, for on-the-road template work, and template user interface design for Viz Ticker 3D and Viz Trio. See Working Without a Database Connection.
- **diff_tool:** The path to a text compare tool executable can be entered here. This will enable the possibility to compare script files when conflicts between local scripts and repository scripts appear.

Viz

The following settings define interaction between the Viz Pilot system and Viz Engine.

- **onairpwd:** Sets the Register ON AIR password for Director users.
- **data_timeout:** Sets the Viz Engine connection timeout in milliseconds for receiving data from Viz Engine. Default is 60000 milliseconds (60 seconds). This is useful if you are loading a scene that is very complex or loads a big amount of data that will take some time to render (for example a map scene).

Viz IO

Viz IO (input/output) specific settings.

- **enabled:** Enables the Viz IO connection. Y or N.
- **host:** The machine hostname (or IP) of the Viz IO machine.
- **port:** The Viz IO communications port.

- **auto_reconnect:** In case the connection to the Viz IO is cut, this flag indicates whether to automatically try and re-establish the connection. (It will attempt after 30 seconds, then after 1 minute, then after 2 minutes, then after 4 minutes.)
- **offset:** Used to offset the incoming triggers by a fixed number.

5.4.2   Configuration Files

The following configuration files are installed with the Viz Pilot system.

- The default files are stored in the program folder with names in the format `xxx.default.xxx`.
- Any changes are stored to files in `%PROGRAMDATA%\Vizrt\Viz Pilot x.y\Config.`

File	Use
components.cfg	Default Component Palette settings for Template Wizard
custom_components.cfg	Custom Component Palette settings for Template Wizard
extra_components.cfg	Custom Component Palette settings for Template Wizard
keyboard.kbd	Keyboard Configuration and Shortcuts for Director
vtw_keyboard.kbd	Keyboard Configuration and Macros for Template Wizard

5.5　Database Configuration

Most features of the Viz Pilot system require a database connection, usually an Oracle database. However, Viz Pilot 7.0 and later support using Graphic Hub as a database.

- Oracle Database
- Graphic Hub Database

5.5.1　Oracle or Graphic Hub?

- Oracle is designed for performance when handling large amounts of data. Graphic Hub is designed for ease of use and low maintenance costs.
- There are more options for fine-tuning and customization with Oracle. Graphic Hub works out of the box.
- The main reason for using Graphic Hub as the database is ease of setup and maintenance. Viz Pilot systems typically have a Graphic Hub. Moving the database into a Graphic Hub rather than an Oracle database means there is one less component to install and maintain.
- Graphic Hub comes with its own tools for maintenance tasks, such as importing and exporting data, backup and replication. With Graphic Hub there's no need to keep a dedicated database administrator.
- Object Store is not supported by Graphic Hub.

See also

- Working Without a Database Connection

5.6 Oracle Database

This section describes how to manually configure the Oracle database connection:

- Oracle Database#Oracle Database Configuration During Installation
- Oracle Database#Database Setup Options
- Oracle Database#Database Initialization File Configuration
- Oracle Database#Database Registry Settings
- Oracle Database#Database Service Names and SIDs
- Oracle Database#Database TNS Alias
- Oracle Database#Database Client

5.6.1 Oracle Database Configuration during Installation

Configuration

The Oracle database needs to be configured in the following places:

1. For client applications, use the Changing Settings either during or after the Viz Pilot System Installation. See Configuring the database connection.
2. For Pilot Data Server, use the installation interface during Pilot Data Server Installation.
3. For Media Sequencer, in Director go to Tools > Media Sequencer Configuration > Database Tab.

Workflow Options

> ⚠️ **Note:** For existing installations with Oracle, it is recommended that Media Sequencer be directly connected to Oracle.

In Viz Pilot 6.x and earlier, the Media Sequencer must be connected directly to Oracle for change notifications and to fetch Pilot data.

In Viz Pilot 7.x, an alternative setup has been introduced, where Pilot Data Server exposes all the data needed by Media Sequencer through its REST interface. This setup also provides Change Notifications (see Pilot Data Server Administrator Guide) using the Stomp protocol. It is possible to configure Media Sequencer to connect to Pilot Data Server instead of the Pilot database.

> ⚠️ **Note:** This setup requires Media Sequencer 4.0 or later.

5.6.2 Database Setup Options

There are several ways to configure the database connection settings. The following sections describe the configuration options for Director and Viz Pilot News. Configurations for Director also apply Template Wizard and Object Store.

Initialization File

In this case, ContentPilot.ini is used by all the Viz Pilot system components, except Viz Pilot News, which uses registry settings.

Since it is possible to connect to the Viz Pilot database by **only using an INI file**, the `tnsnames.ora` file may be omitted.

If the database username and password is changed (default is *PILOT*/*PILOT*), they too must be set under the *Database* section of the ini file. See Database Initialization File Configuration below.

Initialization File and Registry

The combination of an initialization (INI) file and registry settings allows the use of a TNS alias. This is useful if the database setup is used by more than one application.

For more information see Database TNS Alias.

Initialization File and Environment Variable

An INI file and an environment variable allows the use of a TNS alias.

However, this could potentially affect the use of aliases for other database clients, see Database TNS Alias. If more than one *tnsnames.ora* file is used, it is recommended that the files are merged when setting the *TNS_ADMIN* as an environment variable. If this is not desirable, add the *TNS_ADMIN* to the registry.

Registry

Only Viz Pilot News can solely use database Registry Settings.

5.6.3 Database Initialization File Configuration

Set the Database Properties for the INI File

1. Start the Initialization File Editor.
2. Select ContentPilot.ini, and click **Open**.
3. Select the **Database** section and enter the following parameters:
   - **login**: Database login string (*PILOT*).
   - **pwd**: Database password string (*PILOT*).
   - **utf8**: Sets Director encoding for all database communication. Set Y (Yes) to use the recommended and default UTF-8 encoding, or N (No) to use the local Oracle client's encoding setting.
   - **nls_lang**: Define the NLS_LANG environment variable here.
   - **name**: Database name (either the Database TNS Alias or the SID).
     The SID can be combined with a hostname `//<hostname>/<SID>`. This is useful since in most cases the database host is not the same host as for the client applications.
     When using a *tnsnames.ora* file, use the TNS name (default: *VIZRTDB*).

5.6.4   Database Registry Settings

If connection properties are entered during installation, Registry Settings will already be set for each application.

5.6.5   Database Service Names and SIDs

An Oracle System ID (SID) is a name that uniquely identifies the database instance, whereas a service name is the Database TNS Alias that is given when users remotely connect to the database. The service name is recorded in the *tnsnames.ora* file on the client. The same name can be used for both the service name and the SID, or any other name can be used.

A service name is a feature in which a database can register itself with the listener. If a database is registered with a listener using a service name, the service name can be used as a parameter in the tnsnames.ora file. Otherwise, a SID can be used in the tnsnames.ora file.

Different service names are used for each database instance with Oracle Real Application Clusters (RAC).

A service name specifies one or more names for the database service to which this instance connects. It is possible to specify multiple service names in order to distinguish between different uses of the same database. It is also possible to use service names to identify a single service that is available from two different databases through the use of replication.

SID is a unique database system identifier while a service name is an alias.

If the Viz Pilot system is running without a tnsnames.ora file, all database connection parameters must be specified in the initialization (INI) file. Database configuration can be done during the Viz Pilot System Installation. Reconfiguration can now be done after installation by running the Changing Settings.

> ⚠️ **Note:** Viz Pilot News does not use an INI file, hence the database connection parameters need to be applied during installation. These settings can be edited using the Changing Settings or by manually editing registry settings (however, manual editing is not recommended).

- **Local database:** <SID>
- **Remote database:** <hostname>/<SID>
- **Remote database:** <hostname>/Service name

5.6.6   Database TNS Alias

Oracle's Transparent Network Substrate (TNS) technology provides a network of applications above the existing network of computers. Although TNS technology can be complex to set up, it is useful when more than one machine connects to a particular database, and makes switching over to another database instance more efficient. Switching can be done manually, or automatically as part of a standby database system.

The TNS technology is configured in a *tnsnames.ora* file that can be placed locally, or in a centralized location to manage all of the Viz Pilot system's database connections. The file contains connection information for one or more databases.

In order to make use of TNS, a combination of the Viz Pilot system's initialization file settings and registry settings is used.

This section contains the following topics:

- The tnsnames.ora File

- INI File Settings for TNS Alias
- Registry Settings for TNS Alias
- Setting TNS_ADMIN as a Local Variable

The tnsnames.ora File

The tnsnames.ora file contains client side network configuration parameters and defines incoming database requests. The file also contains all service names and describes each domain name, with protocol, host, and port information.

A tnsnames.ora file maps TNS names to connect descriptors (usually ADDRESS and CONNECT_DATA). The TNS name is usually a shorter and more legible alias for the somewhat cumbersome service name.

⚠ **Note:** A *tnsnames.ora* file can contain more than one TNS name.

A tnsnames.ora file must be created manually when using an Oracle Instant Client. Place the file in a directory with read access.

For other Oracle clients, the default tnsnames.ora__file can be found in the Oracle client folder.

A client needs a connect descriptor in order to connect to an Oracle instance. The tnsnames.ora file can define more than one database so that a backup database can be used in a fail situation. Oracle uses the normal TCP communication between the clients and the database.

- **Local database**: <SID>
- **Remote database**: \<hostname>\<SID>
- **Remote database**: \<hostname>\Service name
  The example below uses two hosts. These settings are required if the database server is set up with a standard Oracle failover mechanism. When failing over a standby solution (not Data Guard) both servers will briefly be unavailable. To avoid disconnect problems on clients, a configuration is required to ensure that clients attempt to reconnect long enough to succeed in reconnecting. This can be accomplished with a connect descriptor as shown in the code block below.

⚠ **Note:** Always check the latest Oracle documentation for current recommendations regarding failover setups.

Example of a *tnsnames.ora* file for a standby failover setup:

```
# Generated by Oracle configuration tools.
VIZRTDB =
    (DESCRIPTION =
        (ADDRESS = (PROTOCOL = TCP)(HOST = SomeHost1)(PORT = 1521))
        (ADDRESS = (PROTOCOL = TCP)(HOST = SomeHost2)(PORT = 1521))
        (CONNECT_DATA =
            (SERVER = DEDICATED)
            (SERVICE_NAME = SomeName)
            (FAILOVER_MODE =
                (TYPE = SESSION)
                (METHOD = BASIC)
                (RETRIES = 180)
                (DELAY = 5)
```

```
            ))
)
```

- **Host**: Preferably host, or an IP address
- **Service name**: Database service name (for example `VIZRTDB.tvchannel.place`)

  The `ADDRESS_LIST` property lists the server addresses. When a server is in standby mode the database is not open and does not register the service name with the listener. This means that if the client tries to connect to the standby server it will be rejected and try the other server instead.

```
(ADDRESS_LIST =
    (ADDRESS = (PROTOCOL = TCP)(HOST = SomeHost1)(PORT = 1521))
    (ADDRESS = (PROTOCOL = TCP)(HOST = SomeHost2)(PORT = 1521))
)
```

The `FAILOVER_MODE` property directs the Oracle client to fail over the session and ignore any currently open cursors. The sub-parameters say that the client must use basic failover (not preconnect) and will retry 180 times waiting 5 seconds between each new attempt (15 minutes) on failover. This should provide enough time for the switch to occur.

```
(FAILOVER_MODE =
    (TYPE = SESSION) (METHOD = BASIC) (RETRIES = 180) (DELAY = 5)
)
```

> ❌ **IMPORTANT!** A failover situation will not be initiated until the user tries to access the database.

INI File Settings for TNS Alias

The TNS alias name must be set in the name property under the Database section in the ContentPilot.ini file (see Database Initialization File Configuration).

> ❌ **IMPORTANT!** In order for the TNS settings to work for Director, Template Wizard and Object Store, the name key must contain the **same** TNS name alias as in the tnsnames.ora file (for example VIZRTDB).

Registry Settings for TNS Alias

The TNS_ADMIN string in the registry contains the path to the *tnsnames.ora* file. This allows all of the Viz Pilot system's clients (Director, Template Wizard, Object Store and Viz Pilot News) to use the *tnsnames.ora* file, and avoids potential conflict with other applications using a local environment variable.

- If you do not have any conflicting database connections there is no need to configure the TNS path. The standard environment variable TNS_ADMIN is the recommended option (this is the standard Oracle setup).
- If you need to set a new/custom path in cases where you are dependent on connections to two different databases, this can be done in the General tab of the Changing Settings, see configuring the database connection. This path is then added to registry and will only be used by the Viz Pilot components. An alternative option is To set TNS_ADMIN as a local variable as described below.

Setting TNS_ADMIN as a Local Variable

Rather than using the Changing Settings, add a local Windows environment variable called TNS_ADMIN, with the same value as the config tool option. The difference is that the config tool will only affect the Viz Pilot system, while the environment variable will affect all applications on the machine using an Oracle client. See also how best to use the Initialization File and Environment Variable.
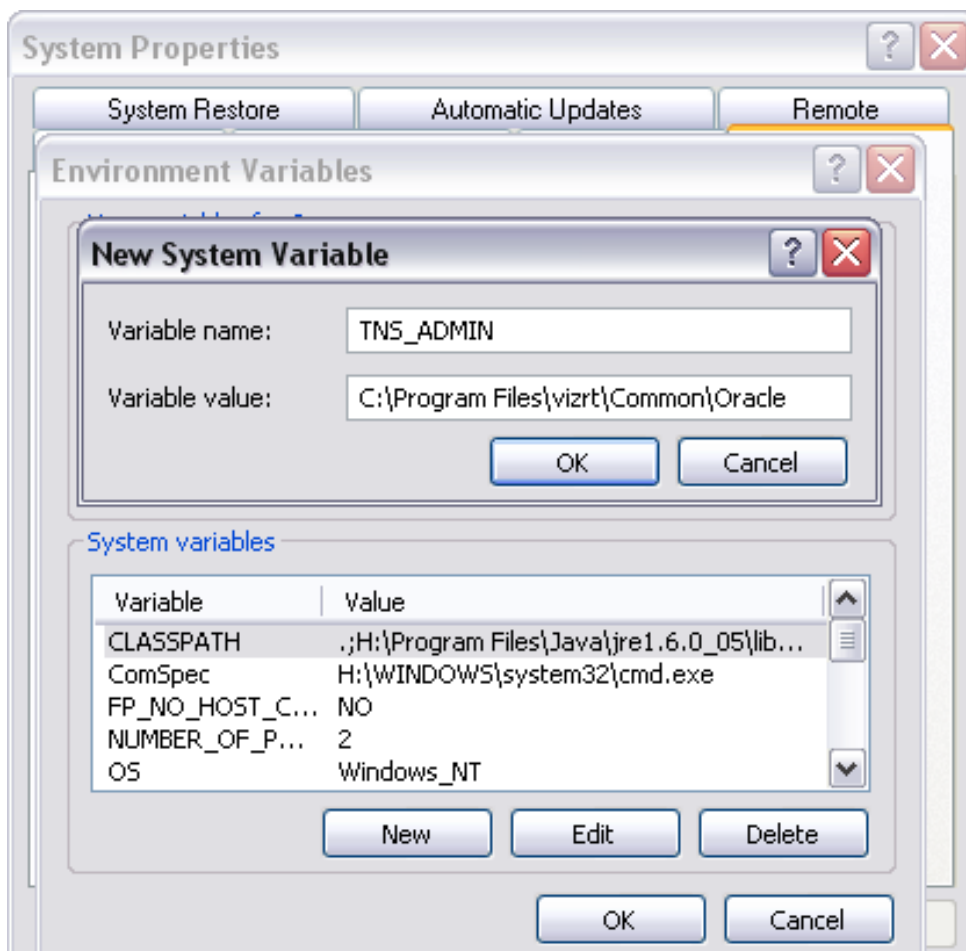
5.6.7 Database Client

The Oracle Instant Client can be selected as one of the components to add when installing the Viz Pilot system. If selected, separate installation of an Oracle client is not required and database configuration for Viz Pilot News can be done during the installation. However, the Oracle Instant Client is only recommended for use with Viz Pilot News, not Director.

It is not necessary to install Oracle's Instant Client if Oracle's Runtime Client is already installed.

> ⚠ **Note:** If Viz Pilot News is embedded in a Java-based newsroom system it is recommended to use Oracle's Runtime Client.

By default, all Viz Pilot system applications use the Instant Client found under " `...\vizrt\Common\Oracle` ". No registry entries are required unless it is desirable to disable the use of the Instant Client and use a full Runtime Client.

> ⚠ **Note:** Media Sequencer must use the Oracle 11g or newer Runtime Client.

Since the Instant Client does not create a tnsnames.ora file, a file must be created if it is to be used. In order to use a tnsnames.ora file, a registry entry or a system environment variable called TNS_ADMIN must be entered.

Using a tnsnames.ora File with Instant Client

1. Open Registry
2. Search for the registry key
    - Windows 64-bit: `HKEY_LOCAL_MACHINE\SOFTWARE\Wow6432Node\\[vizrt]\viz| content pilot x.y`
3. Add the following string value to **TNS_ADMIN**
    - Windows 64-bit: `%ProgramFiles(x86)%\vizrt\Common\Oracle`
    - This sets the path to the location of the tnsnames.ora file.

Disabling Instant Client and using Runtime Client

1. Open Registry
2. Search for the registry key
    - Windows 64-bit: `HKEY_LOCAL_MACHINE\SOFTWARE\Wow6432Node\\[vizrt]\viz| content pilot x.y`
3. Add the integer value **IGNORE_IC** with `1` as parameter.
    - If set to 1 the application will ignore the Instant Client (if installed) and only use the Runtime Client (if installed).

Overriding the Default Oracle Client

1. Open Registry
2. Search for the registry key

- Windows 64-bit: `HKEY_LOCAL_MACHINE\SOFTWARE\Wow6432Node\ [vizrt]\viz| content pilot x.y`
3. Add the following string value **ORACLE_HOME**:
   - Windows 64-bit: `%ProgramFiles(x86)%\vizrt\Common\Oracle`
   - ORACLE_HOME will be added as an environment variable, and the Oracle client this path points to will be used.
4. Add the following string value **OCIDLL**:
   - Windows 64-bit: `%ProgramFiles(x86)%\vizrt\Common\Oracle\oci.dll`

- The OCIDLL string can be used to specify which library the application must use.

> ❌ **IMPORTANT!** An override must only be set if the Viz Pilot system requires a different version of the Oracle client than other programs residing on the same machine.

5.7   Graphic Hub Database

This section describes how to manually configure the Graphic Hub database connection, and contains the following topics:

- Graphic Hub Database Configuration
- Troubleshooting Pilot Database on Graphic Hub

Graphic Hub comes with its own tools for maintenance tasks, such as importing and exporting data, backup and replication, automated cleanup tasks for Viz Pilot data elements and the Viz Pilot change log, and an integrity check. For more information, see the Graphic Hub User Guide.

5.7.1   Graphic Hub Database Configuration

Configuration

The Graphic Hub database needs to be configured in three places:

1. For client applications, use the Changing Settings, either during the Viz Pilot System Installation, or afterwards.
2. For the Pilot Data Server, use the installation interface during Pilot Data Server Installation.
3. For the Media Sequencer, in Director go to Tools > Media Sequencer Configuration > Database Tab.

> ⚠ **Note:** For a detailed procedure, see the Viz Pilot Database Management section in the Graphic Hub User Guide.

Workflow

In **Viz Pilot 6.x and earlier**, the Media Sequencer connected directly to Oracle for change notifications and to fetch Pilot data.

From **Viz Pilot 7.x** ,an alternative setup was introduced. The Pilot Data Server now exposes all the data needed by the Media Sequencer through its REST interface. It also provides Change Notifications (see the Pilot Data Server Administrator Guide) using the Stomp protocol. It's possible to configure the Media Sequencer to connect to the Pilot Data Server instead of the Pilot database.

> ⚠ **Note:** This setup requires Media Sequencer 4.0 or later.

5.7.2   Troubleshooting Pilot Database on Graphic Hub

When using a Graphic Hub as the Pilot database:

- Graphic Hub and Graphic Hub REST are both needed. They install separately and must both be running.
- The Media Sequencer, the clients, and the Pilot Data Server all need network access to Graphic Hub REST.
- The Media Sequencer needs access to the Pilot Data Server as well if it is configured to go directly to Oracle.
- In addition to the existing log files, there is also one specifically for the Graphic Hub client in client applications (Viz Pilot News, Director, Template Wizard) and the Pilot Data Server. The log is located in

%PROGRAMDATA%/vizrt/Logs/GraphicHub and %PROGRAMDATA%/vizrt/Viz GH Pilot. For more details, see the Graphic Hub User Guide.

- The Status Indicators in Director indicates the status of the Media Sequencer's database connection (bottom right). However, the indicator may be green even if the Media Sequencer is connected to a different database than the Director (for example Director is connected to Graphic Hub, while the Media Sequencer is connected to Oracle). If this occurs, an error will occur and the configuration must be changed.
- Media Sequencer 4.0 or later is required for Viz Pilot 7.x. If you are connected to an older Media Sequencer, the Media Sequencer configuration's Database tab in Director looks the same as prior to Pilot 7 (see Media Sequencer 3.x and earlier).

⚠️ **Note:** For troubleshooting on the Graphic Hub side, see the Graphic Hub User Guide.

5.8   Database Parameters

Use the Pilot Data Server to configure settings for the Viz Pilot system's applications. The affected applications usually need to be restarted for changes to apply.

You can access all database parameters from the Pilot Data Server's VCP Parameters page (`http://<hostname>:8177/app/DataServerConfig/DataServerConfig.html`).

> ⊗ **Caution:** Settings should only be changed by administrators.

5.8.1   Viz Pilot Database Parameters

- **activex_preview_timeout:** Sets the time in milliseconds that Viz Pilot News waits for a remote preview Viz Engine connection before continuing. Default is 10000. Can be overridden by the local registry setting PreviewTimeOut.
- **activex_preview_socket_timeout:** Sets the time in milliseconds that Viz Pilot News waits for a local preview Viz Engine connection before continuing. Default is 10000. Can be overridden by the local registry setting PreviewSocketTimeOut.
- **ActiveXShowClipsTab:** Enables or disables the Clips tab for Viz Pilot News. Allowed parameters are 1 or 0. Default is *On* (1).
- **ActivexShowStillsTab:** Enables or disables the Stills tab for Viz Pilot News. Allowed parameters are 1 or 0. Default is *On* (1).
- **ActivexShowTemplatesTab:** Enables or disables the Templates tab for the Newsroom Allowed parameters are 1 or 0. Default is *On* (1).
- **allow_usage_data_collection:** Indicates whether consent was given for usage data collection. By ticking this checkbox, you allow all Vizrt applications connected to this server to collect anonymous usage data, as described in http://telemetrics.vizrt.com/policy.html.
- **app_server:** Sets the Pilot Data Server URI.
- **ax_add_dbdata_to_mosexternal:** Adds extra information about the data in the element to the MOS item.
- **ax_add_itemchannel:** Add default channel for data elements to the generated MOS xml.
- **ax_dataelement_timer_enabled:** Sets the default behavior for showing or hiding the Graphic Event Timing options in Viz Pilot News when Saving an Element. When enabled (1) it will display the timing editor, and when disabled (0) it will hide the timing editor. This setting can be overridden by enabling the *ShowGraphicEventTiming* setting for the Template Information Component in Template Wizard on a template-by-template basis.
- **ax_disable_clear_layers:** Setting to disable the clearing of all layers as the first commands sent when sending to preview
- **ax_disable_data_overwrite:** Disables the ability to overwrite existing data elements in Viz Pilot News and Pilot Edge.

  > ⚠ **Note:** If this setting is enabled, a recurring cleanup task for the database must be introduced to keep the number of data elements and the change log table size under control.

- **ax_disable_dragdrop:** Disables drag and drop in Viz Pilot News. Useful when using a newsroom system that does not facilitate drag and drop interaction with plugins.
- **ax_disable_media_drag:** When enabled (1), this disables the dragging of clips and stills from the media search tab to the rundown. Default (0), i.e. dragging from the media search tab is allowed.
- **ax_disable_overlay_saving:** When enabled (1), this disables Viz Pilot News from creating overlay timelines, i.e. when a clip is opened from the media tab, there will not be any previously saved graphics on the timeline. Overlay saving is enabled by default (0).
- **ax_dont_fetch_thumbnails:** If enabled, Viz Pilot News will not fetch thumbnails for templates and data elements when in list view mode, only in "show icons" view.
- **ax_enableMediaSendToRundown:** Enables (1) the Add to Rundown context menu option when adding media elements to the rundown. This is only useful if the newsroom system does not support drag and drop operations.
- **ax_enable_refresh_button:** Show the refresh button in the Viz Pilot News external preview form. Will refresh the preview without resending the data to the preview server.
- **ax_force_detached_pvw:** Force detached snapshot-preview window in Viz Pilot News/Director.
- **ax_hide_data_elements:** Enabling this will hide the data element list in Viz Pilot News.
- **ax_hide_media_tab:** Setting to hide the media tab in Viz Pilot News.
- **ax_include_mosobj_tag:** Will add a mosobj node to the generated MOS xml. Needed for some newsroom integrations, such as Dalet.
- **ax_mos_edit_save_returns:** Enable this to automatically return to the template list after editing a MOS item.
- **ax_MOSID:** Possible override of the MOSID given in the mositems produced by the Newsroom Component. Default is PILOT.
- **ax_preview_host:** Hostname of the Viz Engine that generates snapshot-preview images and fetch icons of Viz resources (images/geoms/materials, etc.). Also used as fallback vizhost for the ThumbnailGenerator.
- **ax_preview_img_protocol:** Lets you request snapshots in different image formats.
- **ax_preview_port:** Viz Engine port (50008) used to generate snapshot-preview images, and fetch icons of Viz resources (images/geoms/materials etc). Also used as fallback vizhost for the ThumbnailGenerator.
- **ax_preview_tl_legacy_support:** Preview will send values to legacy transition logic location ($other$object) in addition to normal behavior.
- **ax_preview_tl_show_next_sub:** "Toggle*continue_noanim" command will be used when previewing transition logic.
- **ax_remove_continue_count:** Setting to disable the sending of continueCount in mosExternalMetadata.
- **ax_remove_objSlug:** Removes objSlug from the generated MOS xml. This is needed when Viz Pilot News is hosted in AvidCommand.
- **ax_required_version:** Required version of Viz Pilot News. If the Viz Pilot News version is lower then this, a warning will be shown.
- **ax_saveas_cancels_edit:** Default behavior is that when you edit an item already in a script and click "Save As New", the item in the script is replaced by the new item. Enabling this setting still creates a new graphic but does not replace the original item in the script.
- **ax_show_hints:** Disable this setting to disable all tooltips in Viz Pilot News. Tooltips may cause the Dalet newsroom system to crash.
- **ax_show_savechanges_dlg:** If enabled, users will be asked if they want to save the currently open data element or template when opening a new one.

- **ax_use_custom_gui_dlg:** Enables (1) the Viz World Map Editor (WME) to be embedded within the user interface of Viz Pilot News. This parameter is by default not set in the database.
- **clip_default_itemchannel:** If a value is set, it is used as the default channel for video clips. The value is used in the itemChannel element in the MOS XML, and will appear in the Viz Trio playlist and in the NCS. This parameter is empty by default.
  If a video or timeline element is reopened from the NCS rundown, the itemChannel chosen in the rundown will not be reset to the default again, even if the element is updated and overwritten.
- **crop_service_uri:** The URL to the image crop service for example, `http://:8178/`).
- **croptool_max_image_area:** Sets maximum size of a cropped image that will be served by the Pilot Data Server. Anything larger will be resized, while still respecting the aspect ratio of the crop. Maximum size applies even if no cropping is done. Image size = width x height in pixels. 0 = disabled.
- **country_language:** Set the language to use for the country list. By default only English is supported.
- **data_server_url:** Hosting URL of Vizrt Pilot Data Server (for example: http://exampleserver.com:8177) or the reverse proxy server pointing to Vizrt Pilot Data Server. May contain base path prefix in the end of the base URL (e.g.: http://exampleserver.com:8177/pds).
- **delete_data_from_activex:** If set to Y (yes) it will enable the user to delete data elements using Viz Pilot News. Default is Y.
- **DELETE_DATA_PASSWORD:** Password protection of data element deletion in the Director client. If empty, the delete data elements dialog will not be password protected.
- **disable_built_in_vos_search:** Disable the built in VOS search. This should be used to prevent duplicate hits when moving to the new RESTVOS search.
- **graphic_hub_url:** Host or reverse proxy URL of the Vizrt Graphic Hub API used to provide scenes (e.g: http://exampleserver.com:19398). May contain base path prefix in the end of the base URL (e.g: http://exampleserver.com:19398/pilot).
- **image_order_uri_template:** Placeholder URL template for image orders.
- **image_share:** Specifies the path the Resource Panel (for templates) should use when saving a cropped image (`<UNC or Windows path>`)
- **legacy_unicode_handling:** When enabled, Pilot applications revert to pre-8.2 Unicode handling. In 8.2, Unicode support was extended some errors related to handling of more exotic characters, like smileys and other pictograms, were fixed. However, existing template scripts may rely on legacy functionality that stopped working with this update. Enable this option if this affects you. Note, however, that this will also reintroduce some of the issues that were fixed in version 8.2. If possible, the best thing to do is to update all template scripts so that they are compatible with the changes in 8.2. Most importantly this means removing all UTF8Text property usage in scripts and replacing them with code that reads and writes the Text property directly. Ask your support representative for more help if needed.
- **live_update_interval:** Specifies the interval used by Template Wizard's Update Script and the Media Sequencer, to periodically send updates to the update service while on air. The minimum value is 2 seconds. See Using the Update Script Editor.
- **MediaSearch_ItemPerPage:** MediaSearch Items per page.
- **mos_legacy_template:** If true, legacyTemplate element is added to MOS xml. This is a workaround used only if this information can not be retrieved from mosPluginId when a MOS Item is sent.
- **mos_plugin_url:** URL of the Pilot MOS HTML plug-in. Director will use this to open elements and templates that don't have a proper legacy form. Will not work with URLs that use redirection (HTTP 302 answer). For

example `http://pds:8177/pilotedge` will redirect to `http://pds:8177/app/pilotedge/`
`PilotEdge.html` , so it is the latter that needs to be set directly.

- **mse_script_runner:** This parameter can be used to set a different script runner for the MSE. Normally not needed. Default value is blank.
- **order_mgt_uri_template:** Uri template used to invoke the order management order form.
- **preview_server_uri:** Sets the URI of the Preview Server. This machine, with Viz Engine and Preview Server installed, is typically identified as your *frame server.*
  Director reads this setting during startup, and uses it to set the Preview Server to be used by the playlist to fetch thumbnails for the elements. The setting will be written to the /config/preview_server/host node in VDOM.
- **script_runner_uri:** The URL to the script runner that will execute "update scripts" to data/payloads.
- **shared_curious_server:** Sets the hostname or IP address of the Viz World Server. It is not configured when set to *<server>*. See also Director's Viz World Preferences preferences.
- **show_playlist_element_index:** If enabled, shows a column in the Director playlist that displays each items index in the list.
- **slim_mos_xml:** Slim down MOS XML by removing optional tags.
- **spellcheck_dict_filename:** Sets the filename for the dictionary file (.dic) and affixation file (.aff) for the spell checker. This parameter is mandatory and by defualt is *en_US*.
  For example, if *spellcheck_dict_filename* is set to *my-special-en_US*, then the spell checker will look for the two files my-special-en_US.dic and my-special-en_US.aff
  Several spell checking dictionaries have been included as part of the Viz Pilot system's installation. The dictionaries are installed in the default location: `%ProgramFiles(x86)%\vizrt\Common\dicts`
  Other dictionaries can be downloaded from OpenOffice; however, these must be UTF-8 formatted. Contact your local Vizrt representative if there is a need to convert and use another dictionary for spell checking purposes.
- **spellcheck_dict_path:** Sets the path to the dictionary files for the spell checker. By default this parameter is empty, which means the default location is used (`%ProgramFiles(x86)%\vizrt\Common\dicts`). If the files are not located under the default location, use a full path, mapped drive or UNC path.

> ⚠️ **Note:** In order to define a non-default location, the following conditions must be met: (1) *spellcheck_dict_path* must end in a path separator; (2) both the .aff and the .dic files must be present in the path; (3) *spellcheck_dict_filename* must be set. Otherwise, the default location is used.

- **template_format_binary:** Saves the template's/data element's binary to database.
- **timeline_editor_proxy_port_range:** Sets the port range (in the format "a-b;c-d;e;f-g") from which the internal HTTP proxy that deals with authentication for the Timeline Editor in Pilot News takes its listening port. If empty, or if this optional parameter is missing, then a random port will be chosen from the ephemeral range, otherwise from the range specified. Multiple ranges can be given, but the total number of ports to try is capped at one hundred.
- **timeline_update_service:** The service document URL for the Timeline Update Service.
- **vcp_schema_name:** for example, PILOT.
- **video_mode:** Video mode for channel. PAL or NTSC.
- **vos_allow_edit_from_dll (optional):** Enables or disables the toolbar in Object Store. When disabled it will disallow registering and editing of images and person information when Object Store is used with Viz Pilot

News and Template Wizard. This option can be added to the database. Values are *Y* for allow edit and *N* for disallow edit.

- **VOS_PASSWORD:** Shows the encrypted Object Store password that is used to enforce access control when using the Object Store Settings window. Is only visible if access control has previously been enabled. This setting can be used to reset the password by deleting it; however, it cannot be used to change the password. See Object Store Settings.
- **vtw_disable_unique_name_check:** Enable this setting to allow two or more templates to have the same name.

5.9   Registry Settings

The table below shows default registry settings and other settings that might be useful for Director and Viz Pilot News. It is recommended to consult a local Vizrt representative or support person before altering registry settings. It is also recommended to test changes before applying them to a production system.

Registry settings that are not covered are either legacy settings and no longer supported, or are settings that must not to be changed. The table below covers settings that can be changed during installation or manually using the Registry Editor.

Start Windows' registry editor (regedit.exe) and open the following path to edit your parameters:

- Windows 64-bit: `HKEY_LOCAL_MACHINE\SOFTWARE\Wow6432Node[vizrt]`

5.9.1   Registry Settings

Key	Name	Description
ActiveX\ DataBroker	DatabaseName	Sets the database name for Viz Pilot News. Supported values are TNS name alias, connection string or a full TNS connection string.
	Password	Sets the database username
	UserName	Sets the database username
ActiveX\ TemplateFiller	BCSDeviceName	Sets the name for the device that is connected to the Avid iNEWS Control Air (formerly known as iNEWS Broadcast Control System (BCS)). Default device name is *cg*.
	BCSTemplateName	Sets the template name. Default is *pilotdata*. This must be configured in Avid iNEWS. See also BCSDeviceName.
	insertBCSTag	Set this to 1 to insert the BCS values in the iNEWS production cue. If using MOS, set this to 0. See also BCSDeviceName.
	mosID	MOS IDs are created by Newsroom Computer System administrators.
	NODB	Disables (1) the database connection. Default is enabled (0). See Working Without a Database Connection.

Key	Name	Description
	PreviewDisabled	Enables (0) or disables (1) both local and remote preview for Viz Pilot News. Default is enabled (0).
	PreviewHost	The preview host is set during installation. This setting will override the database settings, and can be hostname or IP address. For failover, PreviewHost can have more than one host and port number defined.*<host>* defines the Viz Engine hostname for remote preview. If *<host>* is set without a trailing *<port>* then the PreviewPort number must be set. Parameter:<host><host>,<host>,<host> ...<host>:<port>, <host>:<port> ...
	PreviewPort	By default set in the database. This setting will override the database settings. Default ports are 50008 (multiple connections) and 6100 (single connection).*<port>* defines the Viz Engine port number for remote preview.
	PreviewSizeX	By default set by Viz Engine. This setting overrides the aspect ratio set by Viz Engine.
	PreviewSizeY	By default set by Viz Engine. This setting overrides the aspect ratio set by Viz Engine.
	PreviewSocketTimeOut	Viz Pilot News will wait for a socket connection to the local Viz Engine before continuing. By default the "activex_preview_socket_timeout" setting (see Database Parameters) would be set in the database if needed; however, the same setting can also be set in the registry to override the database setting. Recommended default value is 10000 ms.
	PreviewTimeOut	By default set in the database. This setting overrides the database parameter setting for "activex_preview_socket_timeout" (see Database Parameters). Default value is 10000 ms.
	StartBlank	Enables (1) Viz Pilot News to start without fetching the data elements for each template; hence, this setting can be used to load and start Viz Pilot News more quickly. Default is disabled (0).

Key	Name	Description
	use_utf8	Enables (1) or disables (0) the use of UTF-8 when writing to the database. It is NOT recommended to disable UTF-8 if the database configured to use UTF-8. Default is enabled (1).
	UseVCPDM	Enables (1) Viz Pilot News to be used with the Viz Device Manager. Default is disabled (0). See also BCSDeviceName.
Preview Engine	ShowConsole	Enables (1) the Viz Engine console to always be visible when using local preview. This setting will override the user preference. Default is disabled (0).
Viz Pilot	ORACLEHOST	Oracle database hostname. Note that INI file settings will override registry settings.
	ORACLESID	Oracle database SID. Note that INI file settings will override registry settings.
	TNS_ADMIN	Alternative to a full connection string. If a TNS name file is configured, setting the path may be used to set a different location than Oracle's default location, and to avoid conflicts with similar database connections defined as a local environment variable.

6   Launching Pilot

Once the relevant Viz Pilot Components have been installed and configured, it's time to start Director:

1. Start Media Sequencer.
2. Configure Director's Database parameters in the local initialization file to connect to the Viz Pilot database.
3. Double-click the icon on the desktop, or
4. Select the program from the **Start** menu:
   (**All Programs > Vizrt > Viz Pilot x.x > Director x.x**)

7   Newsroom Integration

Newsroom integration is usually handled by:

- *Viz Pilot News*, an embedded client that runs within the newsroom system client.
- The Quick CG tool used to make cg-like elements, which can also be installed on newsroom system clients.

Both are used by journalists to fill templates with content and store them as data elements. The data elements are then put into a running order/rundown that is created using a newsroom client and made available to Director for monitoring and playout through Viz Gateway or Viz Device Manager.

7.1   Viz Pilot News

Depending on the features selected during the installation, Viz Pilot News lets you search for templates, still images, audio and video clips. Templates can be opened and saved as data elements. Stills and videos can be searched for and added to templates. However, they can also be added directly into the newsroom story for a fullscreen playout. Graphics and graphics with video can be previewed locally or remotely. For remote preview, low resolution video clips are used instead of high resolution video clips.

7.2   Quick CG Tool

The Quick CG tool is simpler and more efficient when creating data elements that only contain text (such as lower thirds).

This section covers the following topics:

- Timeline Editor
- Field Linking and Feed Browsing
- Viz Engine Preview in the Newsroom
- Quick CG
- Table Editor
- Viz Pilot News
- Order Management

7.3   Timeline Editor
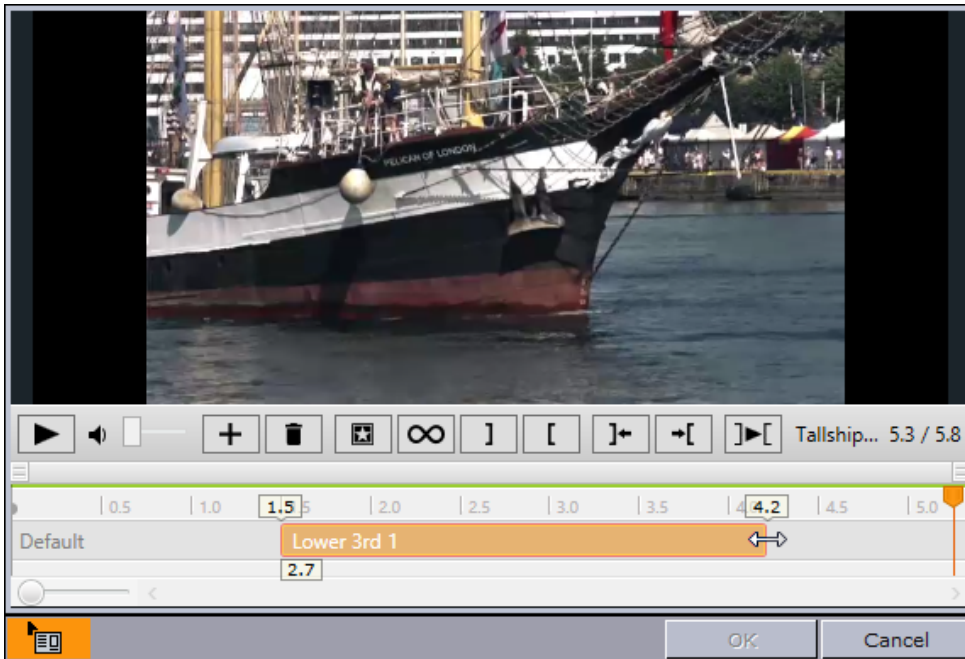
The timeline editor is used to add graphics to a video timeline. The editor is available to Viz Pilot News users.

- Search for a video using Viz Pilot News and preview proxy versions of the video clips. When you have found/ edited the video, you can open the video in the built-in timeline editor.
- The timeline editor offers basic functionality for adding graphics to the video timeline. The graphics are added using data elements that are saved to the Viz Pilot database (normal graphics workflow).
- When using data elements, the timeline editor will add them as placeholders on the video timeline. The placeholders will display snapshots of the graphics as video overlays (fetched from the Preview Server). The placeholders can of course be manually adjusted on the timeline.
- A story item containing clip and graphics references from the timeline editor will appear as a group in the playlist. Issuing a `Take` command on the group will trigger the video clip and timed graphics.

This section contains the following topics:

- Timeline Editor Functions
- Keyboard Shortcuts for Timeline Editor
- Working with the Timeline Editor
- Timeline Editor Preview Mode

7.3.1   Timeline Editor Functions

This section contains the following topics:

- Editor Control Bar Functions
- Timeline Editor Functions

- Multiple Tracks in the Timeline Editor
- Setting a Timeline Name

Editor Control Bar Functions
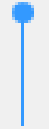
The editor consists of a control bar and a preview window.
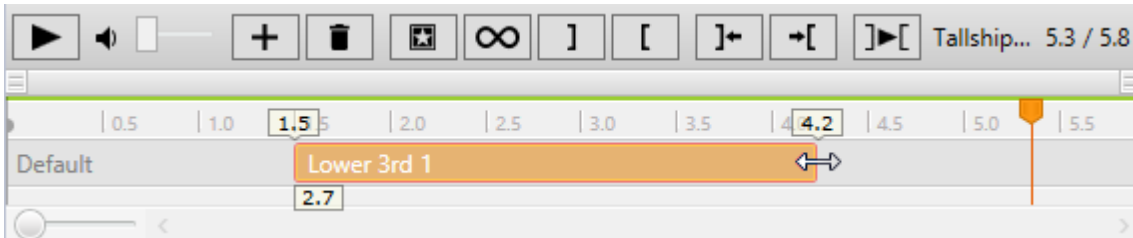
Timeline Editor Functions

The control bar contains the following functions.

Function	Description	Keyboard Shortcut
	Play or **Pause** the clip.	SPACE
	Adjust the **volume** of the clip	
	Add graphics to the timeline.	INSERT
	Delete the currently selected graphics from the timeline.	DELETE
	Scrub marker (cursor). Drag to scrub the clip.	

Function	Description	Keyboard Shortcut
	Click the poster frame marker to **jump** the cursor to the marker. The preview window will display the frame that will be used as the poster frame.	
	Set poster frame to the current position of the cursor.	P
∞	When selected, the video will loop during playout. Note that graphics will not loop.	**CTRL + T**
]	Set the **mark-in** point to the current position of the cursor.	I
[	Set the **mark-out** point to the current position of the cursor.	O
]←	**Jump** the cursor to mark-in point.	SHIFT + I
→[	**Jump** the cursor to mark-out point.	SHIFT + O
]►[	**Play from** mark-in to mark-out.	SHIFT + SPACE
	Drag the **Zoom** control to increase or decrease the time scale shown on the timeline.	Mouse wheel or **CTRL + '+' / CTRL + '-'**
	Set timeline name. The Set Timeline Name button appears when you hover over the video title.	

Function	Description	Keyboard Shortcut
1.0	The Timeline Editor supports progressive download. The **Buffering indicator** is displayed above the timeline - *gray* above timeline sections where buffering has not yet been done, and *green* where the media is buffered locally.	

Multiple Tracks in the Timeline Editor
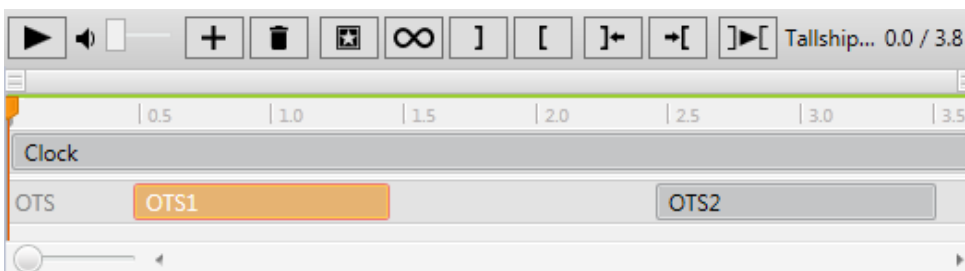
The Timeline Editor supports multiple tracks for transition logic scenes. This allows graphics to overlap and play out correctly, but requires that all data elements are based on the same transition logic set.

The tracks reflect the layers in the transition logic scene. A track is displayed in the timeline editor if an element using that layer has been added.

Graphics from the same layer must not be overlapped, and the timeline editor will indicate a possible conflict by coloring the graphics placeholders red, as shown below:
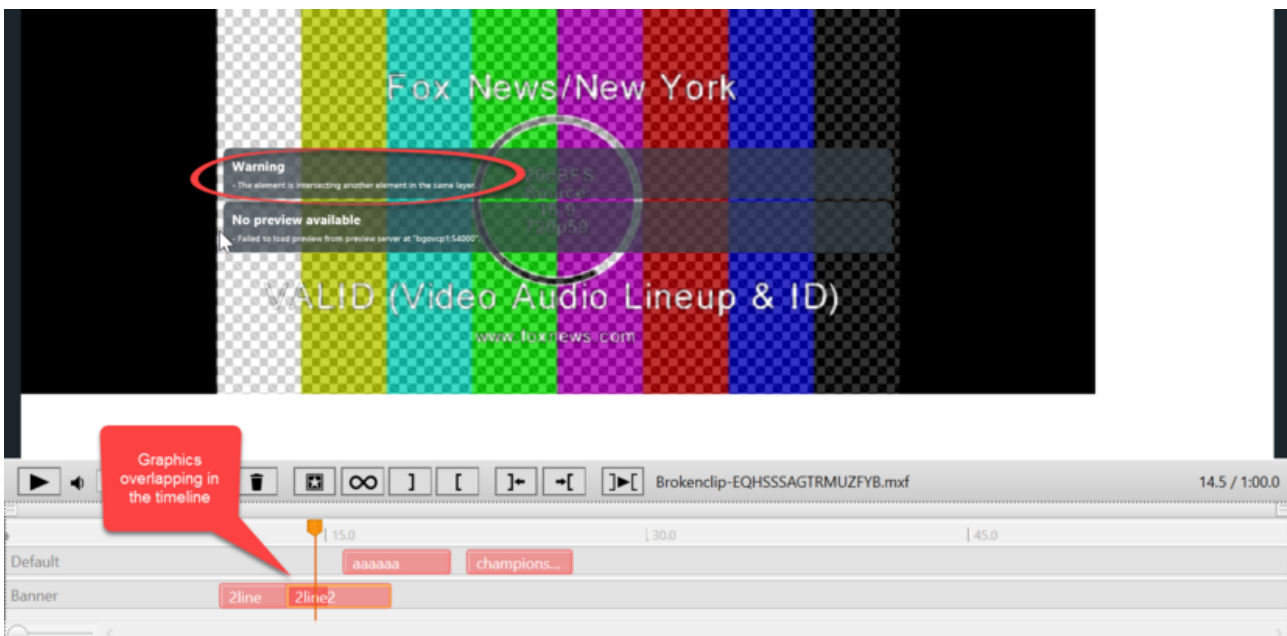
Positioning Elements on the Timeline with Snapping

Graphics snap to each other if they are on the same track.

To reproduce:

1. Open a video with multiple graphics on the same timeline.
2. Drag one graphic towards another.
3. Elements snap together once they get close enough. This means the in and out points on the elements line up on the timeline, as shown here:

> ✅ **Tip:** Press **CTRL** while dragging a graphic to disable snapping.

Setting a Timeline Name

After adding graphics in the timeline editor, you can give a *timeline name* to the resulting composite element. If the timeline name is set, it is put into the MOS element (mosAbstract node), otherwise the video title will be used. The timeline name will be displayed in the newsroom story and in the playlist.

Reopening a MOS item in a newsroom system that supports updating the items, will also update the existing name.

Note that if you open an existing item that has a timeline name, the timeline name will remain the same when editing, but will not appear as a changed name next to the label. This enables you to override the timeline name again or set it back to empty if you wish to use the name that was previously set.

The atom XML generated by the timeline editor will have the timeline name set as the atom:title (this remains empty if not set).

You can:

- **Hover** your cursor over the video title/timeline name to see names in full.
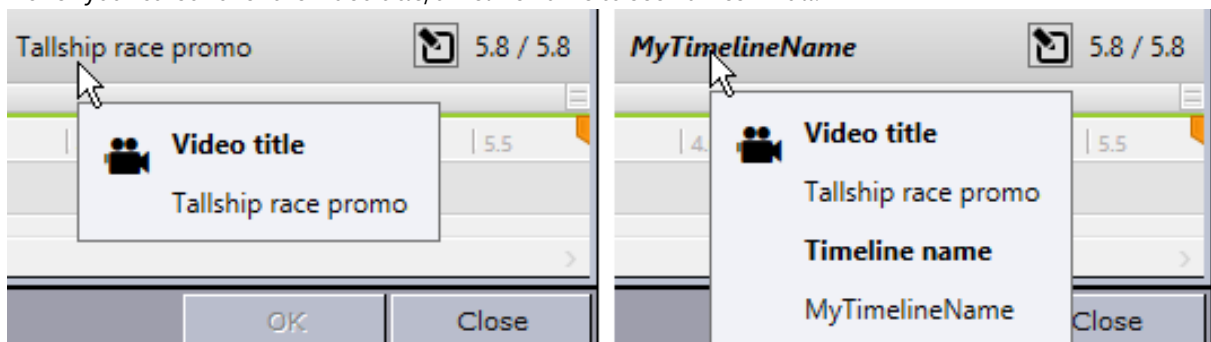
- **Set** or **edit** the timeline name, by either:
  - Right-clicking the video name and select **Set timeline name.**

- Hovering over the video name and pressing the Set timeline name button .
- **Clear** the timeline name by right-clicking the video name and selecting **Clear timeline name.**

7.3.2   Keyboard Shortcuts for Timeline Editor

Key	Description
Mouse wheel	Zoom in and out
CTRL + '+'	Zoom in
CTRL + '-'	Zoom out
CTRL + 0	Reset zoom
INSERT	Add new graphics
ENTER	Open selected data element for editing
DELETE	Remove currently selected graphics from the timeline
SPACE	Play/Pause
ARROW	Move focus to next or previous graphic item
CTRL + ARROW	Move graphics in small steps back and forth on the timeline
CTRL + SHIFT + ARROW	Move graphics in big steps back and forth on the timeline
SHIFT + ARROW	Select multiple graphics
CTRL + ALT + ARROW	Increase or decrease duration of graphic by a small amount
CTRL + ALT + SHIFT + ARROW	Increase or decrease duration of graphic by a large amount
CTRL + L	Move the selected graphics next to the one to the left
CTRL + R	Move the selected graphics next to the one to the right
CTRL + ALT + L	Stretch the selected graphics next to the one to the left
CTRL + ALT + R	Stretch the selected graphics next to the one to the right

Key	Description
CTRL + H	Sets video position to start of the clip
CTRL + E	Sets video position to end of the clip
CTRL + T	Activate looping of the video
,	Scrub backwards one frame. (If frame rate not provided in the asset, scrub back 0.02 seconds)
.	Scrub forwards one frame. (If frame rate not provided in the asset, scrub forward 0.02 seconds)
CTRL + ,	Scrub backwards 1.0 second
CTRL + .	Scrub forwards 1.0 second
P	Set poster frame
I	Set mark-in
O	Set mark-out
SHIFT + I	Go to mark-in
SHIFT + O	Go to mark-out
SHIFT + SPACE	Play the mark-in to mark-out area.
SHIFT + drag the in/out marker bar	Move the bar between the in and out points to move both points at the same time.

⚠️ **Note:** Media keyboard shortcuts for play, pause and mute should also work.

7.3.3 Working with the Timeline Editor

This section contains the following topics:

- Add graphics to a video clip timeline
- Timeline Update Service
- Viewing an Asset Stored in Viz One
- Known Limitations

Adding Graphics to a Video Clip Timeline

1. In Viz Pilot News, click the **Media** tab to search for videos.
2. Right-click the video item, and from the context menu select **Open in Timeline Editor**.
3. To add graphics, click the **Add** button
   - This will open the Template and Data Elements view.
4. Select a template or previously saved data element, and click **OK**.
   - Clicking **OK** will prompt the template or data element to be saved.
   - After it is saved you will see a graphics placeholder on the timeline.
5. Adjust the placeholder to set the desired duration of the animation.
6. *Optional*: Repeat steps 3 and 4 to add more graphics.
7. Drag and drop the timed video and graphics onto your story.
   - : Depicts an item that can be added (by drag and drop) to a newsroom story.
   - **OK**: Saves a reopened story item back to the story.
   - **Cancel**: Cancels any changes.

> ⚠️ **Note:** Graphics may be overlaid on the timeline; however, graphics that use the same layer (front, middle, back or transition logic layer) will not play out correctly.

Timeline Update Service

When a video timeline (standalone video or video composite element) is opened in the Timeline Editor, it will check if a Timeline Update Service (TUS) is defined in the system. If defined, it will POST the playable relative timeline to the Timeline Update Service (TUS).

TUS is called both when opening a video from the results of a media search, and also when opening a video or composite element from a rundown in a newsroom system. In this case, although the TUS might already have run on the video when it was previously created by Viz Pilot, it is still run again.

The TUS is provided by the customer themselves, and is an HTTP service that complies with an API. Viz Pilot will invoke the relevant URL and will expect the result to follow the given protocol.

The API prescribes an **HTTP** service, listening on the **configured URL** for **POST requests** with the **media type Atom Feed** (`application/atom+xml;type=feed`), with content as defined by the **Playable Relative Timeline format**, and responding with content of the **same format**.

- **Set up TUS**:
   - The TUS is provided by the customer.
   - Specifications for the playable relative timeline format can be accessed from the Documentation page of the Pilot Data Server itself.
- **Configure TUS** in the Pilot Data Server:
   - Browse to `http://<Pilot Data Server host:port>/settings#Params`
   - Set `timeline_update_service` to the URL that Viz Pilot is expected to (HTTP) POST Playable Relative Timeline documents to. The response is expected to be a document in the same format with any updates.
- **Use TUS once configured**:
   - Open Viz Pilot News and search for videos. When you preview a video in the Timeline Editor, the courtesy graphic is displayed.

- **Example**:
    - One of the use cases for TUS is adding courtesy graphics. In this case the TUS could return an updated playable relative timeline, with a courtesy graphic added, or return the same timeline it was given if no changes were made. If no value is set in `timeline_update_service`, then no courtesy graphic is added (this is the default behavior).

Viewing an Asset Stored in Viz One

As soon as a new asset is published by Viz One and appears in a search result, it can be opened in the Timeline Editor.

For Viz One assets that have been fully ingested and processed by Viz One, the Timeline Editor will provide a full preview. However, if Viz One is still processing the asset (ingesting the high-res video or creating the proxy video), the Timeline Editor will provide a preview based on the information available to it at the time.

While Viz One is processing the asset:

- The timeline duration reported by the Timeline Editor will equal the high-res video duration reported by the asset, at the time of opening it in the Timeline Editor.
- If no proxy video is available from the asset, the Timeline Editor will open with a blank video preview area, with all video preview related controls (play, pause, volume) disabled.
- If there is a proxy video, but its duration (as reported by the asset) is shorter than the high-res video, then video preview will only be available for the first part of the timeline. Seeking beyond that is possible, but the video preview will not be updated there.
- It is possible to add graphics and in/out points even where there is no video preview. These areas are marked on the buffer indicator line with moving black and white stripes.
- The Timeline Editor sets the out point to the length/duration of the currently available portion of the video. If the high-res is still growing when the timeline is opened in the Timeline Editor, then this out point will be somewhere in the middle of the final video.

> ⚠️ **Note:** When played out, only the part before the out point will be played.

- To see an updated timeline for the asset, reopen the Timeline Editor.

> ⚠️ **Note:** Depending on version and configuration, different Viz One installations may behave differently with regards to the details on which the Timeline Editor is reliant. For more information on Viz One, contact Vizrt Customer Support at www.vizrt.com.

Known Limitations

- Graphics may be overlaid on the timeline; however, graphics that use the same layer (front, middle, back or transition logic layer) will not play out correctly. The timeline editor always indicates a possible conflict by coloring the graphics placeholders red. To use overlapping graphics based on transition logic scenes, see Multiple Tracks in the Timeline Editor.
- Current limitations on the Media Sequencer will cause the following behavior:
    - If you issue a `Take` on a timed group (1), and while that group is played out issue a `Take` on graphics in another layer (2), the Media Sequencer will take out the last taken element instead (2)

when issuing the timed `Take Out` command (for the timed graphics). It is therefore not recommended to take other data elements on-air while a timed group is being played out.

- *Invalid Timeline! The element is not fully inside the area marked by the In and Out points.*
  Timelines which have elements which are outside or intersecting the Mark In/Mark Out points cannot be saved or dragged to the rundown.
  In the image below, the first timeline shows a normal element, while the second timeline contains an element which intersects the In/Out points - the second timeline cannot be saved or dragged.
- If *loading timeline elements* (overlay graphics), or *calling update service* (Timeline Update Service) takes more than two seconds, an **abort** button is shown on the loading screen. Clicking the abort button causes the video to be loaded with a missing or incomplete timeline with overlay graphics, depending on which of the two operations was aborted.
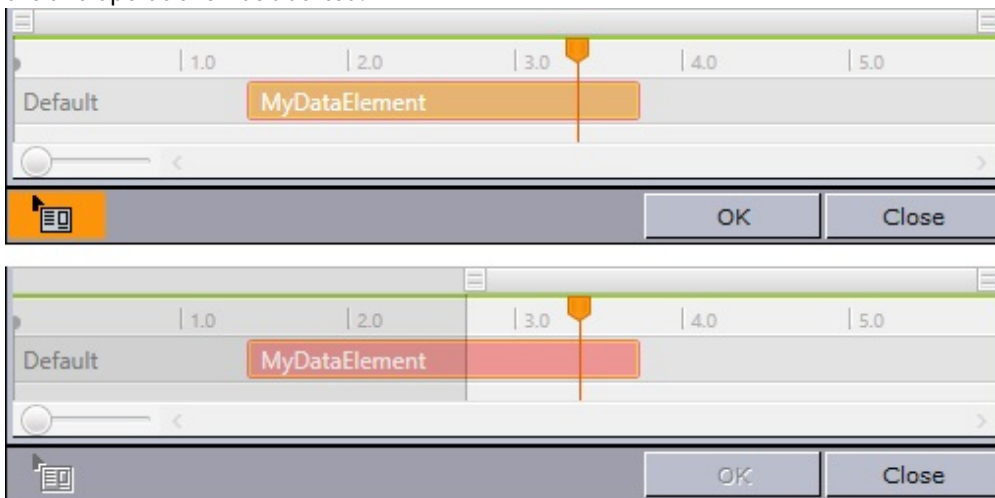
7.3.4 Timeline Editor Preview Mode
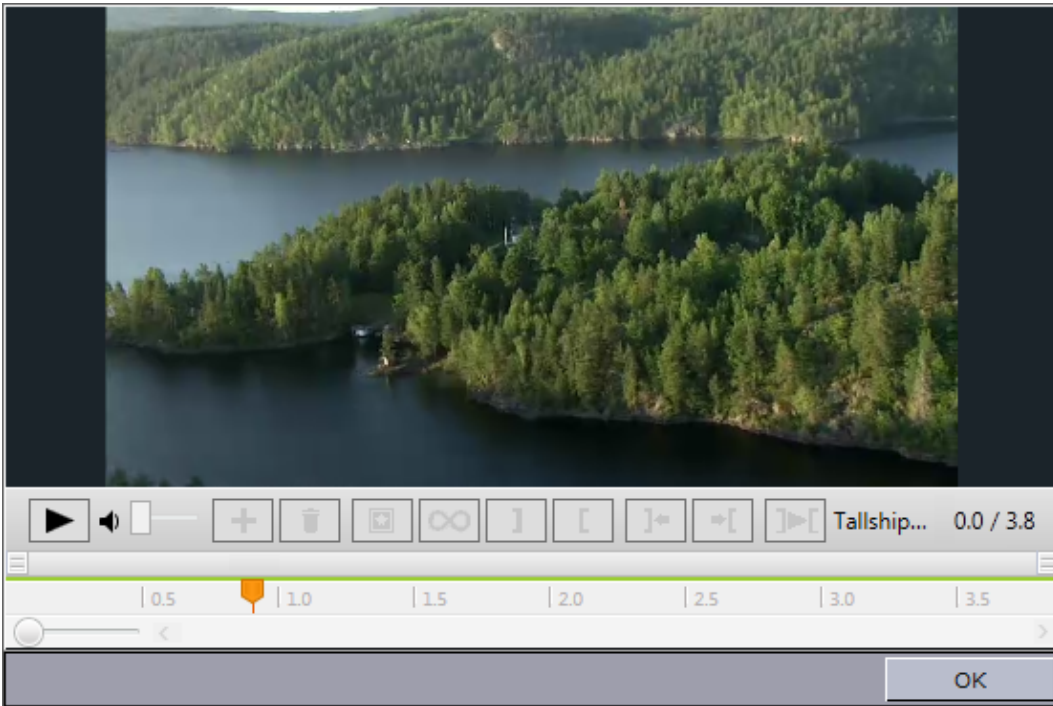
A basic version of the Timeline Editor is available for previewing video elements from within a template or data element. Video preview and scrubbing is available, however the controls for timeline manipulation are disabled.
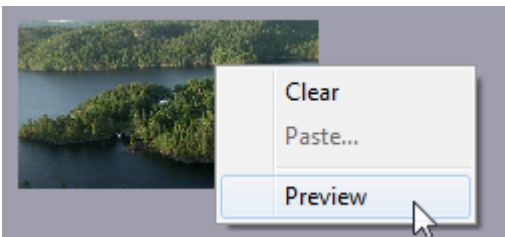
The Preview option is available from the context menu of the video element.

> ⚠️ **Note:** Preview is only available if the image component in the template lists is VideoMediaSearch in it is ImageSources, and is only enabled if a video asset is actually selected.

7.4 Field Linking And Feed Browsing

Viz Pilot supports browsing tab-field values from external sources. Instead of editing tab-field values manually, values from an external feed can be selected. Pick an item from a feed and have the corresponding data applied to fields in the template, rather than entering information manually.

> ⚠ **Note:** Viz Pilot only supports the Atom 1.0 feed standard.

This section contains the following topics:

- Workflow
- Overview
- Feed Browsing

7.4.1 Workflow

Field linking in Viz Pilot is a two-step procedure:

1. **Field Linking:** Linking the fields in a template to various parts of data that will come from entries in a feed. This is a setup step that is usually done once for a template.
   In Template Wizard, create templates using the Feed Linker Component.
2. **Feed Browsing:** After field linking has been set up and saved for a template, the Feed Browser will be available.
   In Viz Pilot News, click on the Feed Linker button to open the Feed Browser window. The Feed Browser lists the items from the feed source defined in the template. Select an item from the list. The data is then mapped to fields in the template.

7.4.2 Overview

Feed URIs can be defined in the control object of the scene design in Viz Artist. Additionally, the feed URI for a tab-field in a template can be defined from within Viz Pilot. Each page then uses the feed browser to edit the tab-field value. The specified feed URI in Viz Pilot overwrites any existing feed URI from the control object. A reimport of the scene does not reset the specified feed URI by the field definition specified in control object.

Tab-fields generally contain several properties, each of which relate to a control plugin in the scene. Properties are grouped under a tab-field by a naming convention.

> ℹ **Example:** The ControlText plug-ins specifying the field identifiers 1.name, 1.score and 1.image will form a tab-field 1 with the properties name, score and image. Selecting a feed item for the tab-field 1 should apply the parts of the selected item to the properties name, score and image at once. The individual property values should be editable as usual. The feed item selection works as a pure fill-in help.

The feed linking is usable across all tab-field types.

> ⚠ **Note:** Make sure you define matching object types, a thumbnail link for control image tab-fields, and so on.

> ⚠️ **Note:** The feed browser supports two types of feeds: flat feeds and hierarchical feeds (folder structure).

Tabfield Grouping

To be able to set multiple tabfield values from the same feed entry, the tabfields must be grouped together. This is done by giving them the same prefix followed by the period character. For example, the following tabfield names will generate two groups of tabfields (`candidate1` and `candidate2`):

```
candidate1.name candidate1.image candidate2.name candidate2.image
```

7.4.3   Feed Browsing
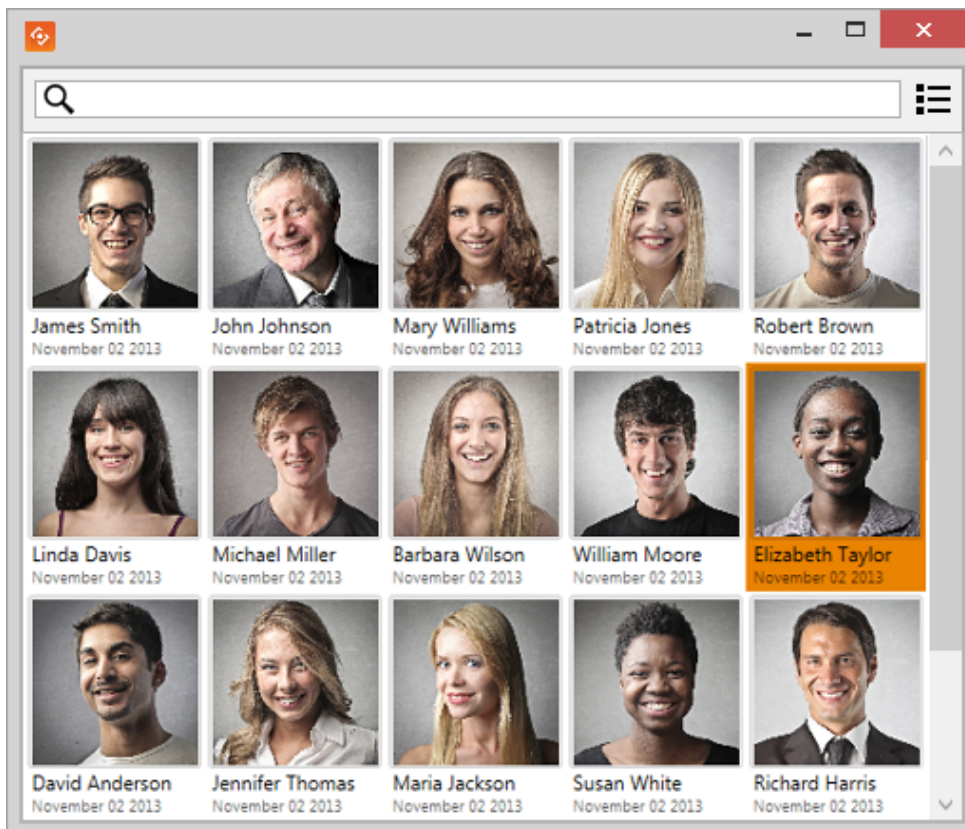
Feed Browser

If they have been given a valid Field Linking URL (see Feed Linker Component), elements can be browsed and selected in the Feed Browser window.

Choose either the thumbnail view or the list view. Right-click the items to Refresh or Sort the items or to Resize/ Zoom the thumbnail view.

Feed Browser Authentication

The feed browser supports basic HTTP authentication. The server must return a HTTP header based on the HTTP standard, like this:

```
WWW-Authenticate: Basic realm="..."
```

The **Pre-authenticated hosts** configured in the Pilot Data Server also apply to the Feed Browser when hosted in a Viz Pilot application (see **Search Providers** in the **Pilot Data Server Administrator Guide**).

Searching and Filtering

If the feed supports OpenSearch, the search box will be enabled and all searches will be done on the server. If not, the search box will be a local text filter box.

7.5   Viz Engine Preview In The Newsroom

Local preview uses a local Viz Engine, while remote preview uses a remote Viz Engine.

- Local Preview in the Newsroom
- Remote Preview in the Newsroom

7.5.1   Local Preview in the Newsroom
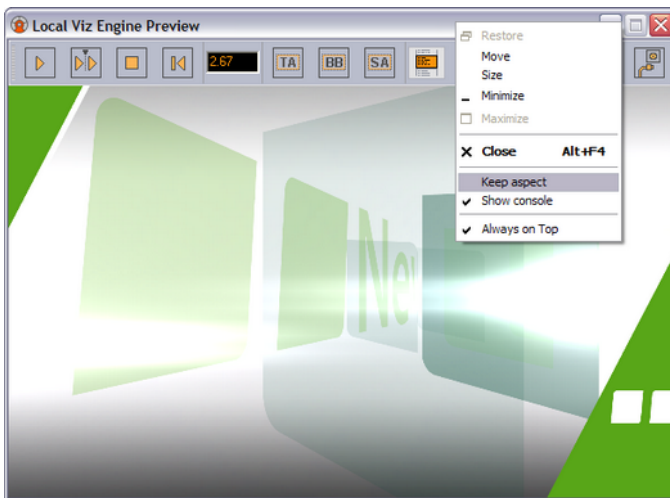
> ⚠️ **Note:** Local Preview requires that the Viz Engine Preview plugin is enabled in the Changing Settings. See Configuring Local Preview.

Properties and Parameters

- **Start:** Animates the scene.
- **Continue:** Continues to animate the scene to the next stop point or completes the animation of the scene.
- **Stop:** Stops the scene.
- **Step Backward:** Steps the scene backwards frame by frame when it is paused.
- **Time field:** Shows the current timeframe for the scene.
- **Title Area:** Shows the boundaries of the defined title area (green rectangle).
- **Bounding Box:** Shows the boundaries of the scene's objects.
- **Safe Area:** Shows the boundaries of the defined safe area (purple rectangle).
- **Show Commands:** Opens a console window that shows the Viz commands. The console window can be enabled by default by adding the ShowConsole registry setting.
- **Restart:** Restarts the local Viz Engine.

Context Menu

- **Keep aspect:** Sets and keeps the correct aspect ratio according to the Viz output format settings when the window is resized.
- **Show console:** Shows the Viz console, showing all commands sent to and processed by Viz.
- **Always on top:** Floats the preview window on top of all other application windows.

7.5.2   Remote Preview in the Newsroom
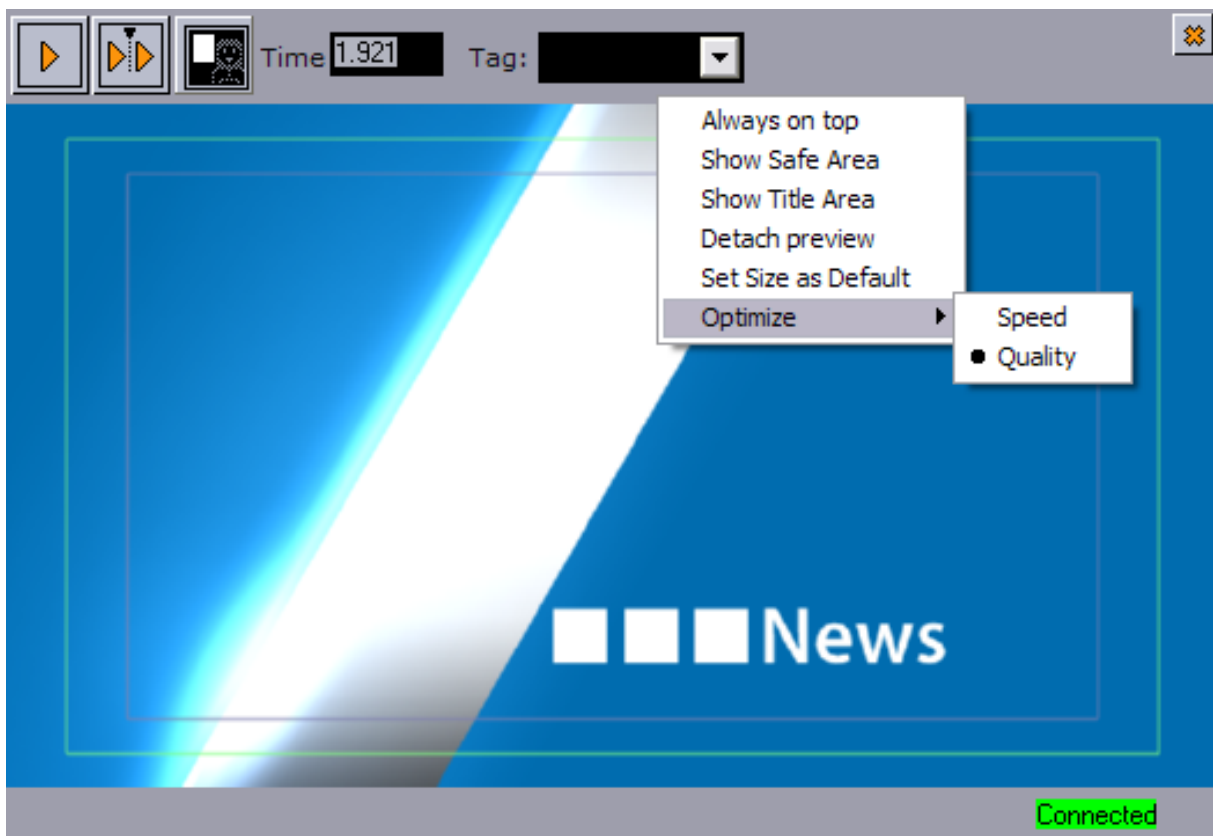
Unlike Remote Preview using Director, Remote Preview using *Viz Pilot News* does not animate the scene. The scene is rendered on a remote Viz Engine and snapshots are sent back to Viz Pilot News according to the time set in the scrub field. Low resolution video clips are used instead of high resolution video clips.

Open the preview by clicking the View icon ⊙ in the Template Editor or pressing **CTRL + SHIFT + P**.

> ⚠ **Note:** To configure Remote Preview, see Configuring Remote Preview for the Newsroom.

Menu Bar

The menu bar contains the following functions:

- **Start:** Asks Viz Engine for a snapshot of the scene from the *first* tag on the default director if the graphic has two or more tags on its timeline.
- **Continue:** Asks Viz Engine for a snapshot of the remaining tags on the default director if the graphic has two or more tags on its timeline.
- **Key:** Shows the key signal of the graphics.
- **Time field:** Shows the current timeframe for the scene. Allows the user to scrub the scene manually.
- **Tag field:** Shows a list of the scene's director tags (not stop tags) on the default director. These tags are used for previewing the scene.
  If the scene contains tags, and fields in the template are linked to these tags, then editing the field will result in the preview jumping to the corresponding point in the animation. For more information on creating tags in scenes, see the section *Working with Events in a Director* in the Viz Artist User Guide.
- **Close (X):** Closes the preview window.

Context Menu

- **Always on top:** Enables the preview window to stay on top of other application windows if the preview window is detached from Viz Pilot News.
- **Show Safe Area:** Shows the boundaries of the defined safe area (purple rectangle).
- **Show Title Area:** Shows the boundaries of the defined title area (green rectangle).
- **Detach preview:** Detaches the preview window from the newsroom system window (see Always on top).
- **Set Size as Default:** Sets a new, customized, default preview window size.
- **Optimize:** Displays the optimization options.
  - **Speed:** Uses an algorithm for calculating the image resolution which results in low quality images that are faster across the network.
  - **Quality:** Uses an algorithm for calculating the image resolution which results in high quality images that are slower across the network.

7.6   Quick CG

The Quick CG (character generator) is a convenient, easy-to-use tool to create an embedded MOS object to be used in a newsroom story.

This section contains the following topics:

- Quick CG Functions
- Working with Quick CG

7.6.1   Quick CG Functions
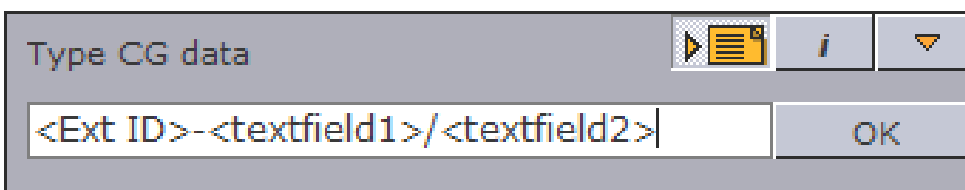
- **Type CG data:** Uses the External Id combined with the data needed to fill the text fields for the graphics. The Media Sequencer interprets the entered values by sorting the *control object fields* of the template alphanumerically according to *control object ID*. It then applies the first value to the first field, the second to the second, and so on.
  Use the delimiters dash (-) and forward slash (/). For example: `200-FirstName/LastName/Designation`.

> ⚠ **Note:** Check for blank spaces in the input. There must not be a space before the dash. A space anywhere else will be included in the value.

- **Auto-paste to Editor:** When selected, it enables the Quick CG to auto-paste the generated XML output to a previously selected editor (for example a newsroom story editor).
- **Settings:** Opens the Settings dialog box to configure another activation key. Available options are; Disabled, Ctrl, Alt, Shift, Home, End, Insert, Num Lock and Scroll Lock.
- **Minimize:** Minimizes the application to the taskbar. Reopen it by double-pressing the assigned activation key.
- **OK:** Initiate the generation of the XML needed to create a proper MOS object that can be added to the newsroom system. If *Auto-paste to Editor* is selected, it will paste the XML directly to the previously selected editor (for example a newsroom story editor).
  The MOS object (XML data) is placed on the *clipboard*.

```
<mos>
<mosItemBrowserProgID>VCPAxFiller.VCPTemplateFiller</mosItemBrowserProgID>
<mosItemEditorProgID>VCPAxFiller.VCPTemplateFiller</mosItemEditorProgID>
<mosID>PILOT</mosID>
<objID>EMBEDDED</objID>
```

```
<mosAbstract> FirstName LastName Designation</mosAbstract>
<createdBy>SomeUser</createdBy>
<created>2007-11-12T13:05:05</created>
<description> FirstName LastName Designation</description>
<mosExternalMetadata>
    <mosScope>PLAYLIST</mosScope>
    <mosSchema>http://www.vizrt.com/mosObj/embedded</mosSchema>
    <mosPayload>
        <embedded_item>
            <external_id>200</external_id>
            <fields>
                <field>FirstName</field>
                <field>LastName</field>
                <field>Designation</field>
            </fields>
        </embedded_item>
    </mosPayload>
</mosExternalMetadata>
</mos>
```

When the MOS object (XML data) is added to a newsroom story editor, it is possible to double-click it and save it as a data element using Viz Pilot News, if required.

⚠️ **Note:** See Variants for how to set External IDs.

7.6.2   Working with Quick CG

Creating a CG Data Element
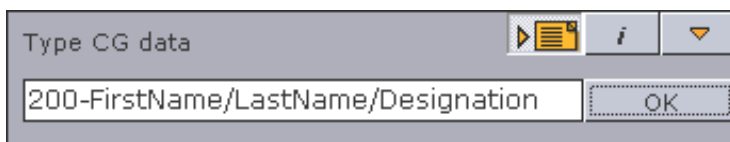
1. Open Quick CG by double pressing the activation key (default is **CTRL**).
2. Enter the template's **External ID** followed by a dash.
3. Enter the **data** separated with a forward slash.
4. Select newsroom story editor and click the **Auto-paste to Editor** button (first button (upper-right)).
5. Click **OK** to add the XML output to the editor, or
6. Paste it manually into the editor by pressing the **CTRL + V** keys.

7.7   Table Editor

If a scene contains a Control List plug-in and the Table Editor Component, a table editor is shown for that tab field when the scene is edited in Viz Pilot News or Director. Changes made in the table editor are immediately updated in the preview window.

7.7.1   Cell Editors

The cells have different editors depending on the data type they display. In addition to text fields, the table editor can show the following special editors in the table cells.

- **Image Cells:** The drop-list shows a list of all images currently available for a specific column. This provides a quick way of choosing images when many cells are to show the same image. The browse button to the right opens the image pool.
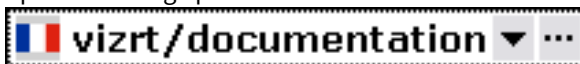
- **Number Cells:** Scenes with exposed numbers, with and without decimals, get a small text editor with a spin button functionality. Click the spin buttons, use the arrow keys, or enter them directly on the keyboard to change the value.

- **Checkbox Cells:** For Boolean values, such as hide and show, a checkbox will be displayed.

7.7.2   Navigation and Sorting

Use the arrow keys on the keyboard to move between the cells.

The table can be sorted manually by clicking on the header bar of the column to sort by. Rows can be rearranged manually by drag and drop. Click on the row once and hold the mouse button to drag and drop it to its new location.

7.7.3   Customizing a Table View

Right-clicking on the table header row opens the Table Editor's context menu, which contains the following items:

- **#:** Shows the row number.
- **Description:** Shows the description field. Descriptions can only be edited in Viz Artist by the designer.
- **<Column names>:** Columns can be hidden or shown by selecting or unselecting the column name.
- **Auto Width:** When disabled the column width can be set manually.
- **Enable Sorting:** Enables manual sorting of the table.
- **Clear Sorting:** Clears any sorting and takes the table back to its initial state.
- **Insert Row (Ins):** Inserts a single row.
- **Insert N Rows**...: Allows for multiple row insertion. The maximum number of rows allowed is set in the scene design.
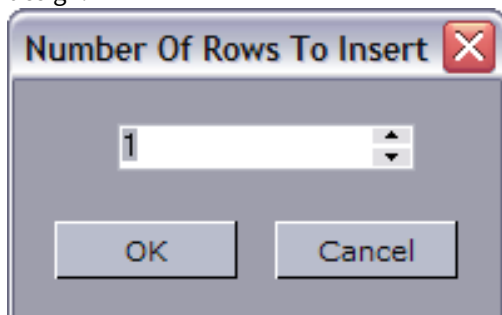
- **Edit Row:** Edit the row data.
- **Delete Selected Rows (DEL):** Deletes selected rows. For multiple rows, select use **CTRL + Left mouse** or **SHIFT + ARROW** keys up or down.
- **Select All (CTRL + A):** Selects all rows.
- **Cut (CTRL + X):** Cuts rows out of the table. Press **CTRL + V** to re-add the row(s).
- **Copy (CTRL + C):** Copies rows from the table. Press **CTRL + V** to add the new row(s).
- **Paste (CTRL + V):** Adds the cut or copied rows to the table.
- **Save Settings:** Saves the column setting for an active template.

⚠️ **Note:** The number of rows that can be inserted or deleted depends on the configuration of the plug-in. See the Chart and List of plug-ins in the Viz Artist User Guide.

7.8   Viz Pilot News

The user interface of Viz Pilot News allows you to fill templates with text, still images, video clips and audio, or simply add them directly to the newsroom system for full screen graphics display.

This section contains the following topics:

- Template Search
- Template Editor
- Working with the Template Editor
- Media Search
- Maps

7.8.1   Template Search

The Template Search tab allows users to search for and open templates, and save them as unique data elements.

This section contains the following topics:

- Template and Data Element Panes
- Functions
- Context Menus
- Search Bar
- Newsroom Workflow without Database Connection

Template and Data Element Panes
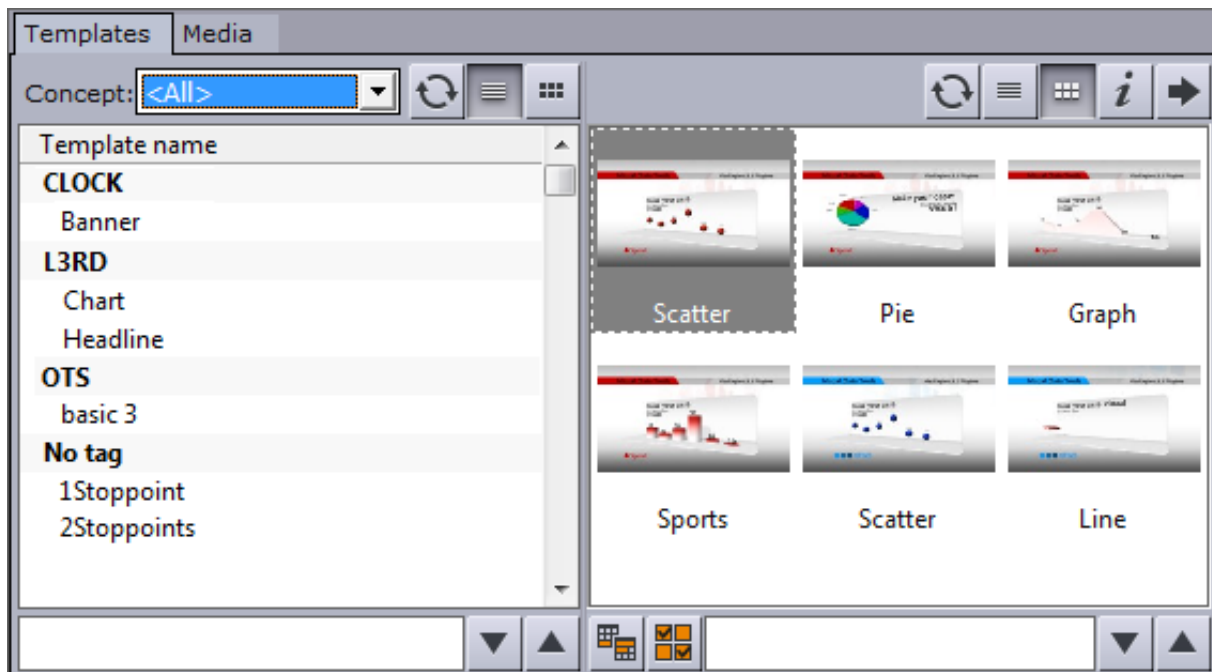
The **Template pane** (left) shows all templates based on the selection made in the concepts drop-list.

To **open** a template, double-click or press **CTRL + SHIFT + O** to open a template by name or **CTRL + SHIFT + I** to open a template by ID.
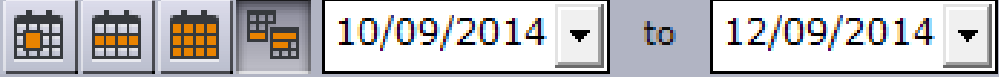
> ⚠️ **Note:** If template tags have been defined, templates are grouped by tag name in the list view. Tags are defined on the **Tag Settings** page of the Pilot Data Server Web Interface (see the Pilot Data Server Administrator Guide). Templates can be tagged in Template Wizard.

The **Data Element pane** (right) shows all data elements based on the selected concept. Click a template to show data elements based on the selected template. Data elements that have been saved as library elements are suffixed with an asterix (*), (see Saving an Element).

> ⚠️ **Note:** Media files that are added to a template are searched for through Object Store and not by using the Media tab.

Functions

Function	Description
<All> ▼	The **Concepts** drop-down shows all template concepts available on the database. Select a concept to see icons for that concept.
🔄	The **Refresh** buttons, for the template pane and data element pane, refresh the window, and reload all templates/data elements available on the database. Keyboard shortcut: **CTRL + SHIFT + R**
▤ ⊞	The **Switch View** buttons switch the template pane and data element pane between list and icon view. When viewed as a list, templates are sorted by Tag name. When viewing all templates (Concept: <All>) generic icons are used.
i	Clicking the **Information** button typically displays version information, the connection parameters used by Viz Pilot News to connect to the Viz Pilot database, and a link to the list of Third Party Component Credits.
← →	The **Arrow buttons** switch between the template editor view, and the template and data element lists. Keyboard shortcut: **CTRL + SHIFT + L**
▦ ▦	When the **Finish** button is clicked, you can select whether or not the data elements view hides data elements that are marked as finished. Keyboard shortcut: **CTRL + SHIFT + N**

Function	Description
	Use the **Calendar** button to filter results based on date range. Select from **Last 24 hours, Last week, Last month, Custom date range:** Keyboard shortcut: **CTRL+SHIFT+D**

Context Menus

Context menu for templates pane:

- **Refresh:** Retrieves new and updated data elements from the database.
- **Copy:** Not available for the template view.
- **Show icon/list view:** Toggle between icon and list view.
- **Font:** Sets the font's type and size.

> ⚠ **Note:** Style and script options will not be saved.

Context menu for data elements pane:

- **Refresh:** Retrieves new and updated data elements from the database.
- **Sorting:** Sets the sorting for the data element view. Options are: Sort by data element name (**CTRL + N**) and Sort by creation date (**CTRL + D**).
- **Show only library elements:** Displays only data elements that are saved as library items. For details, see Saving an Element.
- **Copy:** Copies the data element's XML source so that it can be pasted into a newsroom story item or text editor.
- **Delete:** Deletes the selected data elements.
- **Show icon/list view:** Toggles between icon and list view.
- **Font:** Sets the font's type and size.

> ⚠ **Note:** Style and script options will not be saved.

Search Bar

The search fields are below the template and data element panes. Enter text to search the respective pane for matching templates or data elements.

When searching for templates, Viz Pilot News automatically displays all data elements related to a template. When searching the data elements, the first element matching the search criteria is selected.

Newsroom Workflow Without a Database Connection

The video workflow using Viz Pilot News will continue to work if the connection to the Viz Pilot database is unavailable.

If the Viz Pilot database is unavailable, you *can* :

- authenticate with a previously authenticated host (for example Viz One),
- search for videos,
- add the video to a newsroom story,
- preview the video in timeline editor,
- open existing timeline items and make changes (no preview is available).

If the Viz Pilot database is unavailable, you *cannot* :

- add graphics to video elements using the timeline editor.

> ⚠ **Note:** To attempt to **reconnect to the database**, close and open Viz Pilot News, and then click the **Reconnect** button in the **Templates** tab.
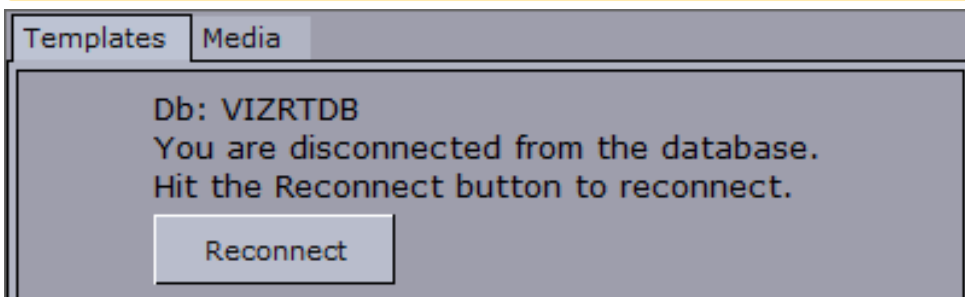
7.8.2   Template Editor

The Template Editor allows data elements to be created from templates, and previewed. The editor is made up of a Control Bar, Template Pane, and Preview Pane. You can set additional properties when Saving an Element. Many of the functions can be performed using the Keyboard Shortcuts for Template Editor.

The image below shows a saved data element with remote preview enabled.
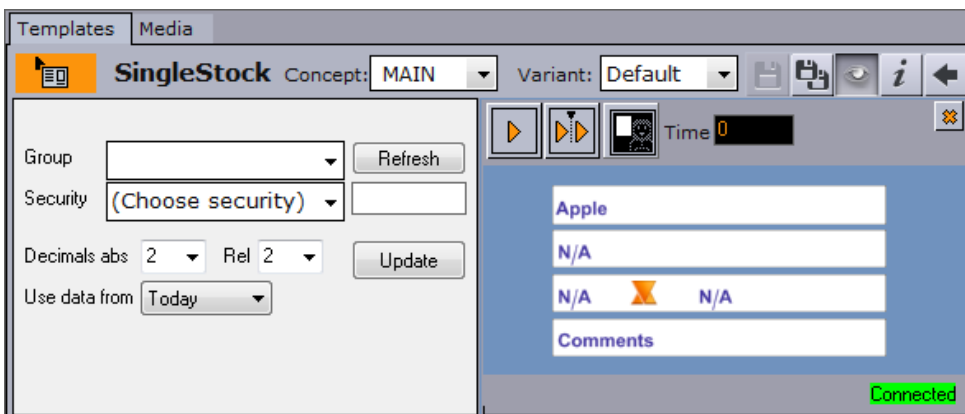
Control Bar

The **control bar** shows the status and the name of the template, as well as the available concepts and variants, save options, preview, information and back buttons.

When the **status** of a saved data element is orange (as above), it indicates that the element has been saved to the database and is ready to be added to a running order/rundown in the newsroom system. Usually a drag & drop operation will add the data element to the news story; however, this may be different depending on the newsroom system.

There are two **save** buttons; **Save** and **Save As** 💾 🗗 . The **Save As** button will show the save dialog and let you create a new data element using the data entered into the currently open template or data element.

Use the **Save** button to save changes to the existing data element that is currently open. In case this is a library element, the save dialog will appear, to help you avoid accidentally changing it. To overwrite it select the **Add to library** checkbox and click **Overwrite**.

The **back** button ⬅ takes you back to the template and data element lists.

Template Pane

The fields and layout that you see in the template pane are designed by template designers, using Template Wizard. It can therefore differ in appearance and contain a variety of elements such as images, buttons, text boxes, map selectors, etc. It's recommended to create an internal reference document for your workplace which describes the templates available for everyone, and how a user should work with them.

Preview Pane

Enable or hide the preview using the **Preview** button 👁 .

When performing preview using a remote Viz Engine, the preview window can be **detached**, giving more space for the template. The local preview window is always detached.

The **context menu** of the preview pane includes the settings for detaching the preview pane, as well as size and optimization settings, and safety boundaries. For details, see Viz Engine Preview in the Newsroom.

Saving an Element
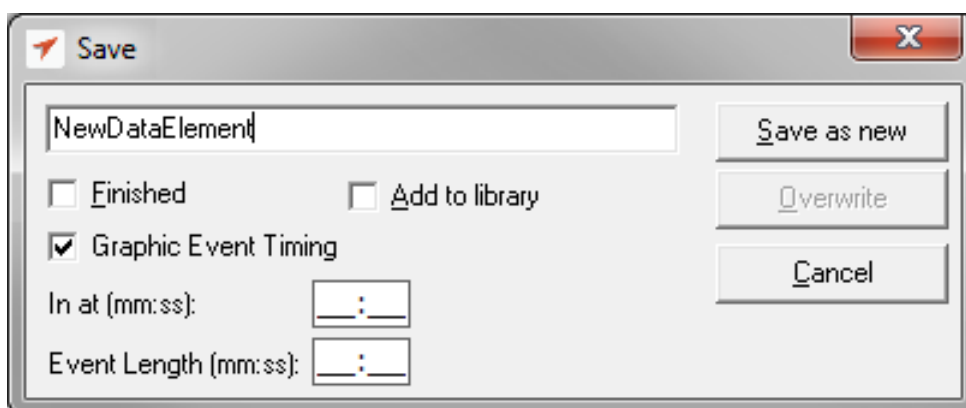
Viz Pilot News has its own save dialog for setting additional properties.

If the Template Spell Checker has been enabled, the save operation will open the spell checker prior to the save dialog. Spell checking is enabled by the template designer using Template Wizard.

- **Finished:** If the data element is made in different revisions and you want to indicate that this is the final one, select the **Finished** check box. This information will be stored in the database and can be shown in the Finished column in the playlist (1 for finished and 0 for not finished). This setting has no impact on the element, other than displaying the value in the Finished column.
- **Add to library:** Select this checkbox if the data element will be used often and you want to prevent it from being deleted. The data element will be saved as a *library element*, and all concept and variant information is kept. This prevents the data element from being deleted in Director when using **Options > Delete Data Items**. Library elements can still be deleted manually, and if the original template is deleted in Template Wizard, all data elements based on the template will also be deleted, including library elements.
  The **Add to library** checkbox will by default be unchecked. To **Overwrite** an already existing library element, select the **Add to library** checkbox, click **Overwrite** and click **Yes** in the confirmation dialog.

> ⚠ **Note:** Data elements that are saved as library elements, will appear with an asterix (*) after their name in the data element list (see Template and Data Element Panes).

- **Graphic Event Timing:** Select this to enable event timing.

> ⚠ **Note:** In order for the flag to be visible, it must be turned on in the Database Parameters (`ax_dataelement_timer_enabled`) or in Template Wizard (Template Information Component > Advanced Properties).

When checked, the following additional settings are available:

- **In at:** Sets a start time (mm:ss), *MOS-itemEdStart*. Timing is relative to the story/group the element is a part of, rather than the complete playlist. The value appears in the *Timecode* column of the (MOS) playlist when opened in Director.
- **Event length:** Sets the duration (*MOS-itemEdDur*) which tells the newsroom system how long the graphics will be on air. The value appears in the *Duration* column of the (MOS) playlist when opened in Director.
  When an element from a MOS rundown is opened in Viz Pilot News, the timing fields in the save dialog will be populated with the timing information from the MOS XML. If the element is saved back to the rundown, the timing information will still be present in the XML.

> ⚠ **Note:** In order for Viz Pilot News to generate the timing information in the MOS XML, the data element must be saved just before the item is added to the rundown. If you have reopened the data element, or you drag it directly from the list of data elements, the timing information will not be available.

Keyboard Shortcuts for Template Editor

Keys	Description
CTRL + ALT + C	Copy a data element from the data element list, or an open element in edit mode.

Keys	Description
CTRL + ALT + S	Saves the data element as a new data element. See also **CTRL + S**.
CTRL + C	Copies a data element from the data element list. See also **CTRL + ALT + C**.
CTRL + D	In data element view you can sort according to a data element's creation date.
CTRL + N	In Template Search you can sort according to template name. In data element view you can sort according to data element name.
CTRL + SHIFT + D	Toggles the **Calendar** button, exposing the start and end date fields.
CTRL + SHIFT + I	In template view you can open a template by ID.
CTRL + SHIFT + L	Switches to the template editor view, and back to the Template Search and data element view.
CTRL + SHIFT + N	Toggles the Finish button, so that the data elements view will only show data elements that are not marked as finished when saved.
CTRL + SHIFT + O	Opens a template by name when in Template Search.
CTRL + SHIFT + P	Toggles the preview. See also Remote Preview in the Newsroom.
CTRL + SHIFT + R	Refreshes the template and data element panes, and reloads all templates/ data elements available on the database.
CTRL + SHIFT + S **CTRL + S**	Saves the data elements. See also **CTRL + ALT + S**.
CTRL + SHIFT + 1	Toggles the image search filter.
CTRL + SHIFT + 2	Toggles the audio search filter.
CTRL + SHIFT + 3	Toggles the video search filter.

Keys	Description
CTRL + SHIFT + 4	Toggles the filter.

7.8.3   Working with the Template Editor

This section contains the following topics:

- Create a data element
- Rename a data element
- Edit a data element
- Add an open data element to the newsroom story
- Copy an open data element
- Copy a data element from the data element list
- Change a concept and/or variant
- Preview a video in a template
- Get the data element's default channel

Creating a Data Element

1. Double-click a template from the Template pane to open it in the editor.
2. Make the necessary changes to the data (e.g. text, images, video clips).
3. Click **Save**.
4. In the Saving an Element, enter a name for the data element.
5. Click **OK**.

Renaming a Data Element

1. Double-click to open a data element in the editor, and click the **Save As** button.
2. In the Saving an Element, enter a name for the data element.
3. Click **Overwrite Existing**.

Editing a Data Element

1. Search for and select a data element and **double-click** it to open it in the editor.
2. Make the necessary changes to the data (e.g. text, images, video clips).
3. Click **Save**.
4. Click **OK**.

Adding an Open Data Element to the Newsroom Story

You have three options for adding a data element to a newsroom story:

1. Drag and drop the orange icon (top-left) onto your newsroom story.

2. Press **CTRL + ALT + C** to copy it to your clipboard and paste it into your newsroom story.

3. Open an already saved data element and drag and drop the **green icon** onto your newsroom story.

Copying an Open Data Element

- Open a data element and press **CTRL + ALT + C** to copy it to your clipboard and paste it into your newsroom story.

Copying a Data Element from the Data Element List

- Select a data element from the list and press **CTRL + ALT + C** (or **CTRL + C**) to copy it to your clipboard and paste it into your newsroom story.

Changing a Concept or Variant

1. In the editor, simply use the **Concept** or **Variant drop-list** to select a new concept and/or variant.
2. Click **Save** to save the data element.

> ⚠ **Note:** Concepts and variants are created using Template Wizard's template manager tool.

Previewing a Video in a Template

Once a video item has been added to the video placeholder:

1. Right-click the video placeholder and select **Preview** from the context menu.
2. The video and any graphics elements that have been added are previewed in the Timeline Editor Preview window.

Retrieving a Data Element's Default Channel

A default channel can be set for a template, which can be passed to the newsroom system when items are added to the rundown.

1. Ensure that a default channel has been defined for the template (see To set the default channel for a template).
2. In Pilot Data Server's VCP Parameters, set the parameter `ax_add_itemchannel` to enabled.
3. Create a data element from the template and drag it to the rundown.

> ⚠ **Note:** To disable this feature, disable the parameter `ax_add_itemchannel`.

7.8.4   Media Search

This section contains the following topics:

- Search and Result Panes
- Media Context Menu

- Details View
- Search and Filter Options
- Filter Media by Person Name
- Media Search Errors

Search and Result Panes
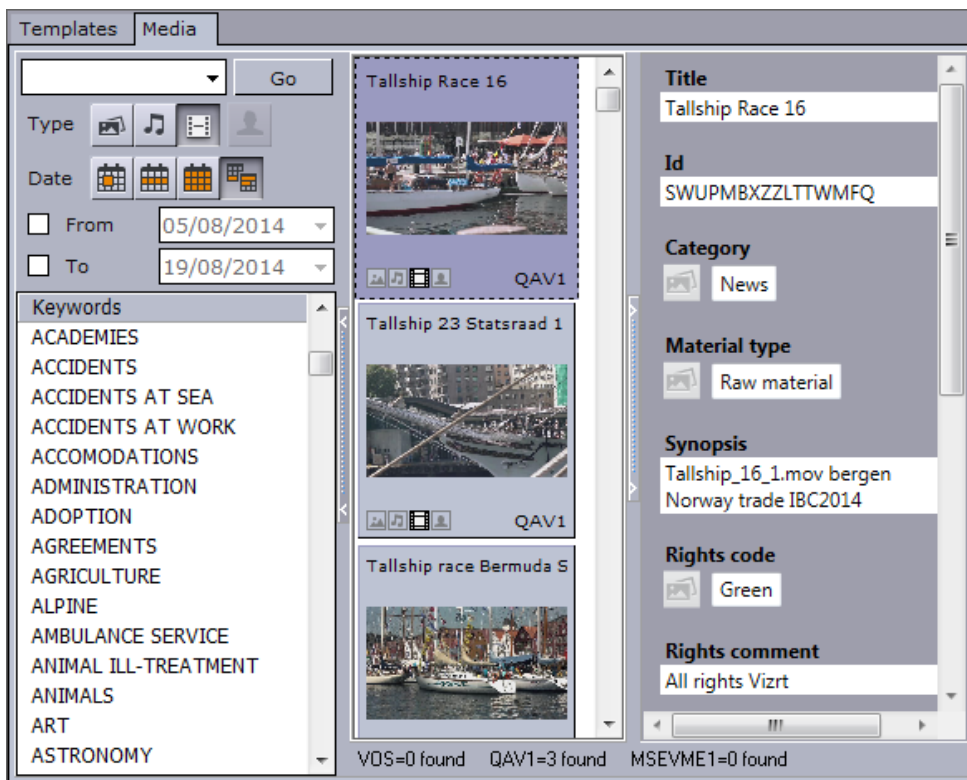
The **Media tab** allows users to search for media assets (i.e. still images and video clips) that can be added directly to the story (e.g. a full screen video clip).

Media information resides on storage systems such as Object Store and Viz One. Object Store traditionally stores still images and person information. Viz One stores video, audio and video stills. The Media tab combines the sources into one.

The left pane shows the Search and Filter Options and a list of categories, while the right pane displays the search results, and optionally, the item details.

In order to reduce network load and to increase the speed of the search, the media search will display the first 15 results from each provider (e.g. Object Store or Viz One) and then ask for more if needed when scrolling. The most recent assets are displayed first. If the number of results is large, it's recommended to refine the search.

If multiple Viz One systems are in use, all of them will be searched and the search results will be combined according to the ID of the clip. i.e. If the same clip is available on multiple Viz One systems, it will only appear once in the search results.

> ⚠️ **Note:** Depending on the data available from each provider, the status bar will display either *ProviderName=N of M shown* or *ProviderName=N found*, where N will increase as more items are fetched. If a provider does not provide sufficient information, the status bar will display *ProviderName=unknown*.

Advanced Asset Search in Viz One

Complex search arguments that can be constructed and syntax details are presented in the table below:

Name / Symbol	Description	Example
AND	Find all items that match *all* given strings	goal AND penalty AND Chelsea This will only return items that contain *all three* words *Chelsea*, *goal* and *penalty*. **Note**: This (AND) is the default operator
OR	Find all items that match *any* given string	dog OR cat This will return items containing *dog*, it will also return items that contain the word *cat*
NOT	Exclude items that match the given string	dog NOT cat This will return items containing dog, but exclude any that also contain cat
-	(Same action as NOT described above) Exclude items that match the given string	dog -cat This will return items containing dog, but exclude any that also contain *cat* **Note**: You must specify a positive term, together with the negation. For example, it is *not* permitted to just write *-cat* as a search argument
*	Wildcard, any string	cat* This will return *cats, catch*, and so on *cat This will return *scat, muscat*, and so on
?	Wildcard, any single character	c?t This will return three-character strings: *cat, cut, cot*, and so on
/	A modifier that extends the search to include stemmed words deriving from the search phrase	"lorry car"/ This will return both *lorry car* and *lorries cars*.

Name / Symbol	Description	Example
Proximity search ()	Also known as sloppy phrase search, allows a matching of terms, within the specified distance.	"lorry car"2 This will return *the lorry contained one car* but will not return *the lorry contained one green car*. "lorry car"/2 This will return *the lorry contained one car* and also *the lorries contained several cars*
""	Search for the literal content held between the quote signs	"*" to search for an asterisk character. "very, clear day" will return only items that contain *exactly* the string *very, clear day*

In addition to simple and complex text searches that you put together, powerful filtering features are already built into Viz One, see *The Viz One Studio documentation*.

Media Context Menu
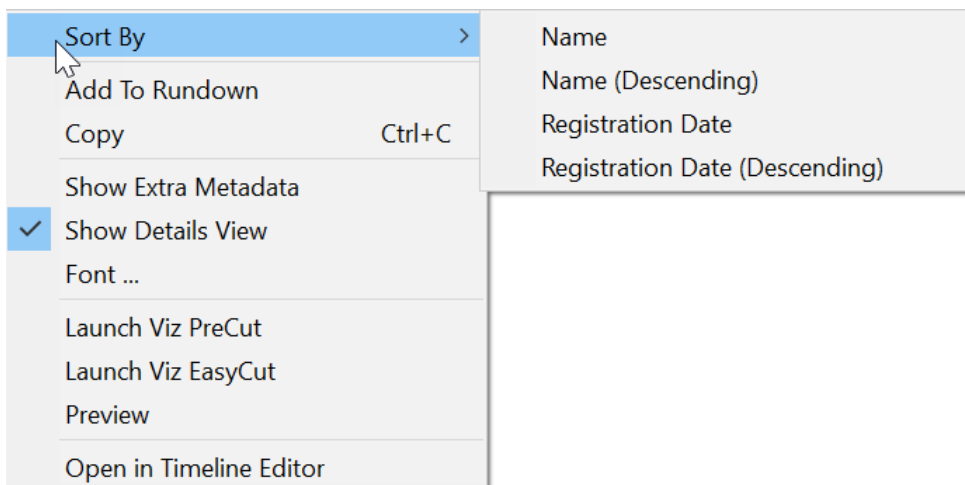
- **Sort By:** Displays a sub menu with sort options.
    - **Name/Name (Descending):** Sorts by name in ascending and descending order.
    - **Registration Date / Registration Date (Descending):** Sorts by registration date in ascending and descending order.
- **Add to Rundown:** Enables newsroom systems, that do not support drag and drop operations, to add media elements to their rundowns. In order to use this menu option, the database setting ax_enableMediaSendToRundown (see Viz Pilot Database Parameters), must be added to the database and enabled.
- **Copy (Crtl+C):** Copy the selected item. It can then be pasted to the rundown.
- **Show Extra Metadata:** Switches the media icons to display metadata such as complete filename, creation date, and clip length and so on.
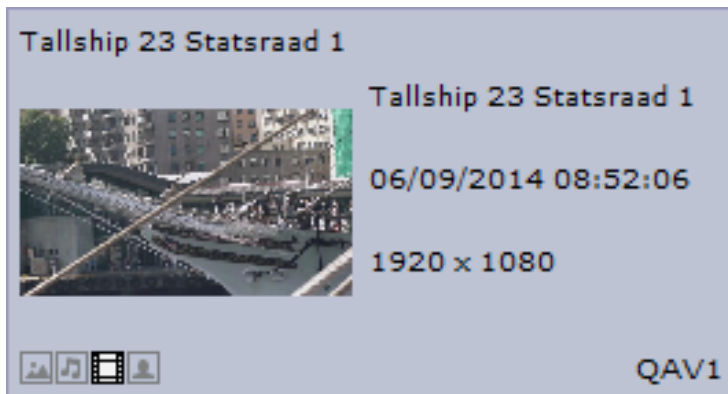
- **Show Details View:** Displays metadata of the selected media item in a separate window. Right-click to view the pane at the bottom or on the side of the Media pane.
- **Font:** Sets the font's type and size.

> ⚠ **Note:** Style and script options are not saved.

- **Launch Viz PreCut:** Opens the selected video clip in Viz PreCut for editing.
- **Launch Viz EasyCut:** Opens the selected video clip in Viz EasyCut for editing.
- **Preview:** Previews images using the Windows Picture and Fax Viewer or the default browser. Is only available for Object Store and Viz One items.
- **Open in Timeline Editor:** Opens the Timeline Editor that enables you to add graphics placeholders to a video clip's timeline.

Details View

Display an item's metadata by selecting **Show Details View** from the context menu in the search results.

The Details View shows the **metadata** that is received from the asset search provider. The type of data available can vary, and can be configured in each asset search provider.

The Details View can be placed at the bottom of the search pane, or to the right. Use the context menu in the Details View to change its location.
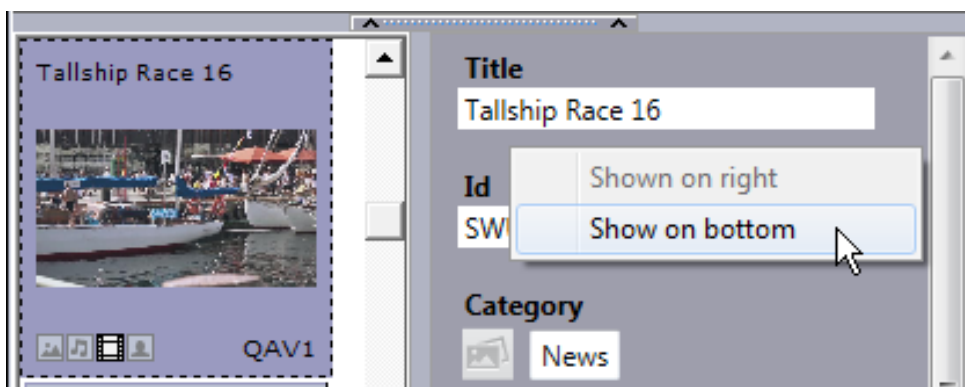
Search and Filter Options

Both Object Store and Viz One searches support free text, keywords, and from and to dates. Combine the use of filters, text search, and keywords to narrow down your results.

- **Search field:** Combo box for entering a search criteria. Previously entered search criteria are remembered per session.
- **Type - Stills, Audio, and Video:** Filters the search result displaying only images, audio or video, according to which filters are enabled/disabled.
  Use **CTRL + click** to select multiple types.

⚠ **Note:** Viz Pilot News does not support the use of audio files.

- **Type - Person:** Filters the search result displaying only pictures that are defined as person (i.e. profile) images in Object Store. See also Filter Media by Person Name.

⚠ **Note:** The **Person Filter** button is only available if a RestVOS Search Provider is configured. If using standard Object Store, search for person name by typing in the Search field.

- **Date:** Filter results based on date range. Select from **Last 24 hours, Last week, Last month, Custom date range**. Filters the search result based on registration dates.
  For Viz One searches, the default date used to search is the CreationDate (date the item was created in the Viz One), however this can be configured to use other dates from the metadata, for example PublicationDate. For more information on how to configure the search, see the *Viz One Administrator Guide*.
- **From/To:** When the **Custom date range** button is selected, the From and To dates are presented.
  IA red background indicates that the from date specified is later than the to date.

- **Keywords:** Filters the search result based on keywords. Templates that use the RequiredKeyWords property will have the relevant keywords pre-selected in the list. Keywords are configured for each Service Provider. See Required Keywords. A warning is displayed if the image selected isn't valid.
  If multiple Viz One systems are in use, all of them will be searched and the search results will be combined according to the ID of the clip. i.e. If the same clip is available on multiple Viz One systems, it will only appear once in the search results.

Filtering Media by a Person's Name

1. Click the **Person Search** icon.
2. Enter part or all of the person's name in the **Search field.**
3. Click **Go.** The search results are grouped by person. The number of items found for each person is shown below the thumbnail.
4. Click the thumbnail of the desired person to view all of the images of that person.
5. Click **Back** to select a different person or to search again.

> ⚠ **Note:** The Person Filter function is only available if RestVOS Search Providers (see the *Pilot Data Server Administrator Guide*) are configured. If using standard Object Store, search for person name by typing in the Search field.

Media Search Errors

- **Startup Error:** An error occurred during startup. The host (or Viz Pilot News) will need to be started again to clear the error.
- **Search Error:** An error occurred during a search. Try doing the search again.

7.8.5   Maps

Viz Pilot News also integrates with Viz World, which allows you to add animated and/or still maps to your graphics. See, the *Viz World User Guide*.

7.9   Order Management

This section contains the following topics:

- Order Management in Viz Pilot News
- Sample Fields for Work Orders
- Submitting a Work Order from Viz Pilot News
- Order Status and Warnings

7.9.1   Order Management in Viz Pilot News

The Order Management System is designed to close the gap between the newsroom users ordering content and the department fulfilling these requests. The Order Management System is built on top of Viz One, Vizrt's media asset management solution.

If Order Management is configured, you can submit an order if you cannot find the media you need from a Media Search. See the **Setting Parameters using Pilot Data Server** section in the Pilot Data Server Administrator Guide.

Order Management comprises three levels - the first two are managed in Viz One, and the third is submitted from Viz Pilot News:

1. **Projects** are the top level of a work order system. Examples of a project are *National Elections*, a single show called *The Evening News*, or an entire facility. Typically a management role creates and manages Projects.
2. **Requests**: A Project may have one or more Requests associated with it. An example of a request could be a *set graphic, a lower-third, a highly stylized video*, or a *Viz Scene*.
3. **Work Orders** are the lowest level and where tasks get done. This is where media is filed, approved or rejected. One or more work orders can be associated with a Request.
An example of a work order is all variants of a lower-third required for the project. Work Orders are generated by templates inside the Newsroom plug-in. Templates can contain one or more clip/image selectors.
Users are notified via iNEWS Top Line Message and/or email as their requests are processed.

7.9.2   Sample Fields for Work Orders

Name	Description
Mandatory Values	
Requestor	User name of the requestor. This can be overridden and does not have to be the same as the user creating the project.
Project *	Name of the Project this work order shall be associated with.
Request *	Name of the Request that this work order shall be associated with.
Media title	Title of the Request.

Name	Description
Send notification	Enables/disables notifications for the Request.
Optional Values	
Description	Additional information about the Request.
Air date/time	Date and time the work order airs.
Contact information	How the user fulfilling the request can contact the requestor. For example: *call me in edit bay 3* or *I am out of office, use alternate contact*. Information not available in the corporate directory typically goes here.
Also notify	Other people who will be notified as the Request is processed. A semicolon separates multiple email addresses.

> ⚠️ **Note:** *Projects and Requests must have MOS Enabled set to *Yes* to appear in the Newsroom plug-in drop-down.

7.9.3   Submitting a Work Order from Viz Pilot News

1. Open a template or data element which has an image component
2. Click the image component to open the Media tab
3. Perform a search and inspect the results. If you cannot find a suitable asset, click the **Order** button below:
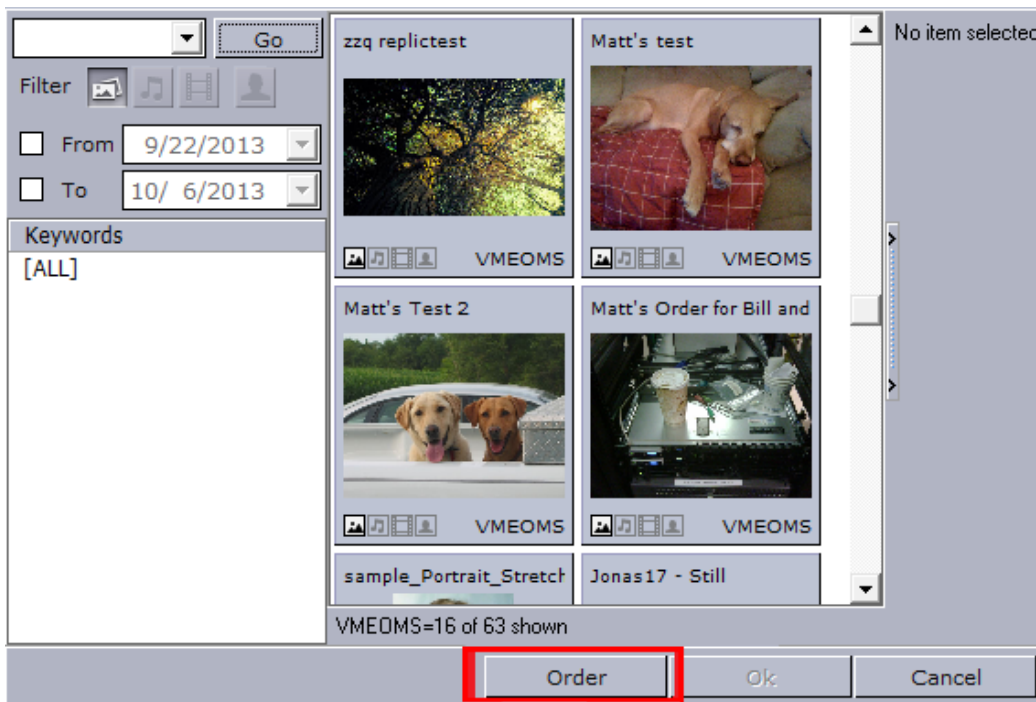
4. If prompted, log-in to the system.

5. In the **Order** menu, fill in the mandatory fields at minimum (indicated with a star *) and select either **Submit** or **Submit and Attach File(s)**.

⚠ **Note:** Users in Viz Pilot News create Work Orders, while Projects and Requests are configured in Viz One.

> ⚠️ **Note:** The Order Form interface window can appear smaller, hiding the Submit button. To avoid this, add the following code in the *Sub InitForm*:
>
> ```
> Set objShell = CreateObject("WScript.Shell")
> objShell.RegWrite "HKCU\Software[vizrt]\ActiveX
> 8.6\TemplateFiller\MayamOrderManagement\Width", 510, "REG_DWORD"
> objShell.RegWrite "HKCU\Software[vizrt]\ActiveX
> 8.6\TemplateFiller\MayamOrderManagement\Height", 640, "REG_DWORD"
> Set objShell = Nothing
> ```

6. If you selected **Submit** and **Attach File(s)**, browse for files in the **Order** window. You can optionally add comments about each file in the **Comment** column. When ready, click **OK**.
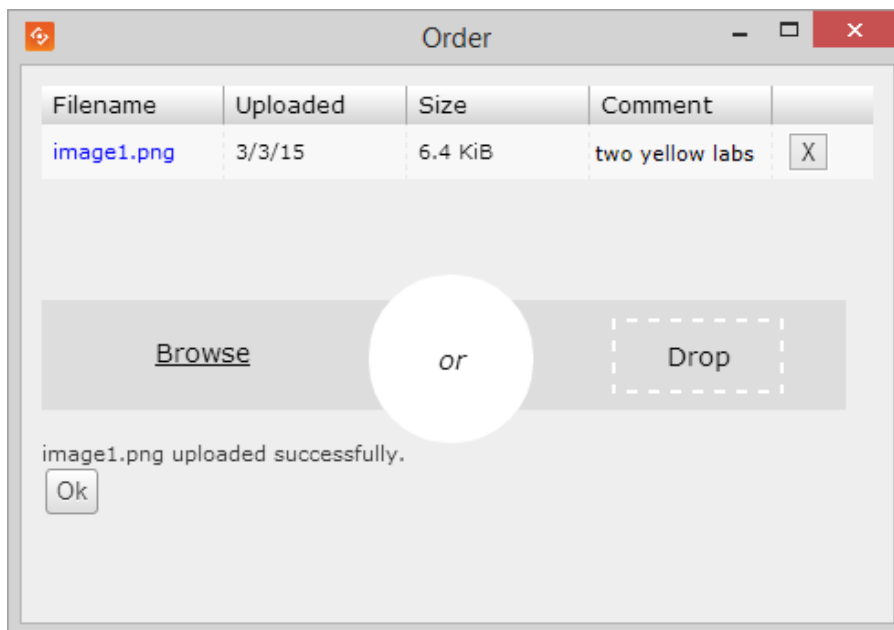
7. Back in the **Templates** editor, the media selector icon changes to **Order**
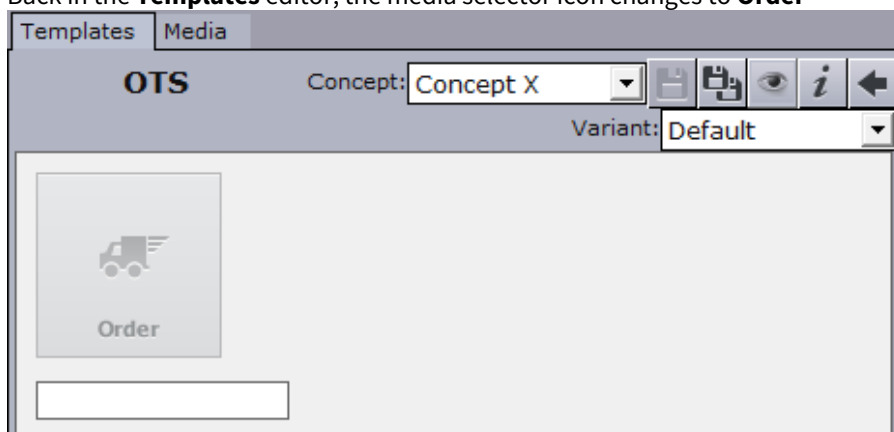
8. Save the template and insert it into the newsroom rundown. Once the order is fulfilled, any material needed to air this data item is automatically associated.

> ❌ **IMPORTANT!** When re-opening a saved data item where a work order has been placed but *not yet fulfilled*, the Order icon is *not* displayed. Instead, a red border stating that the media is invalid is displayed.

In previous releases, an asset was not considered to be fully ready and could not be cropped or displayed until an order was both fulfilled and approved (fulfilled-and-approved). Starting with Viz Pilot 6.0, fulfilled-and-approved will not be handled in the same way. If you create a data element with a fulfilled-but-not-approved asset, a yellow warning border is displayed but the item can be cropped and displayed. However, if you later reopen the same data element, the system no longer identifies it as an order (since it has been accessed); consequently, the warning border will not appear even though the order remains unapproved.

7.9.4   Order Status and Warnings

From left to right in the image above:

- **No order submitted** (Get Image placeholder)
- **Submitted** (Order placeholder)
  On pending items, you can right-click and select **Open in Studio** to open a browser with information about the order, including name and ID.
- **Fulfilled** but not yet approved (yellow warning notification)
- **Approved**
- **Invalid** (red error notification over placeholder), for example *File not found.*
  The status of an order is refreshed when a data element is opened. Hover over the warning/error notifications to read the full message text.

> ⚠️ **Note:** If a video thumbnail from Viz One fails to appear with a red border around the placeholder and the following error message "*Refreshing a media order failed with the error: Failed to load the asset from the URL "http://<Viz-One>" (The remote server returned an error: (403) Forbidden).*", go to the Pilot Data Server search provider configuration (see the **Setting Parameter** section in the Pilot Data Server Administrator Guide) and provide a Fully Qualified Domain Name for the Viz One hostname.

8 Director

Director is Viz Pilot's control room application.
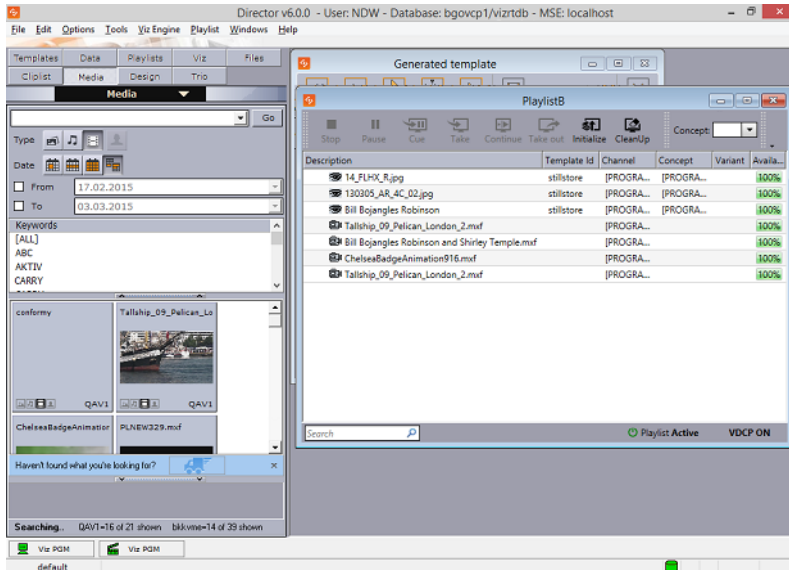
This section contains the following topics:

- Director Configuration
- Keyboard Configuration and Shortcuts
- Director Preferences
- Director Tools
- Viz Engine Preview in Director
- Status Bars
- Resource Panel
- Templates
- Data
- Viz
- Files
- Trio
- Media
- Playlist Window
- Cliplist
- Playlists
- Director Main Menu
- Export and Import in Director

8.1　Director Configuration

This section describes how to configure Director:

- Command Line Startup Options
- Media Sequencer Configuration
- Configuring the Pilot Edge Window
- Working with Stillstore Images
- Profile Configuration

8.1.1   Command Line Startup Options

Startup Options

Media Sequencer is used to create, store and play out data elements. Although Media Sequencer can be installed on a separate machine, it is recommend to run it on the same machine as Director.

For failover or backup, you can specify which Media Sequencer the Director must use; it is recommended that a unique desktop shortcut is created for each Media Sequencer that can be used.

Edit the properties of each shortcut, and add the following:

- `-mse <hostname>`

    In case you need to switch from one Media Sequencer to another, it is useful to have the same profile setups on each Media Sequencer so that fail situations are easier to handle.

Command Line Options

The following command line options are supported by some or all of the Viz Pilot system's applications. Command line options are added to the program's target path.

- **All:** Director, Object Store and Template Wizard
- **Dir:** Director
- **TW:** Template Wizard

Command	Default	Apps.	Description
-db <dbName> -u <dbUser> -p <dbPassword>	VIZRTDB	All	Specifies the Oracle database to connect to. dbName can be either a TNS alias or connect string. Uses the TNS alias VIZRTDB if nothing else is specified. ⚠ **Note:** Do not use both -db and -gh at the same time.
-gh <URI> -u <dbUser> -p <dbPassword>		All	Specifies the Graphic Hub database to connect to. ⚠ **Note**: Do not use both -db and -gh at the same time.

Command	Default	Apps.	Description
-disable_auto_open_recent _playlists		Dir	Disables automatic loading of the last opened playlists on startup.
-inifile <filename>	localhost	Dir TW	Director and Template Wizard support both full path references and single file names. The latter assumes that the ini file is located in the %ProgramFiles% folder or %PROGRAMDATA% \Vizrt\Viz Pilot x.y\Config\.
-nodb		Dir	Run Director in NODB mode (see Working Without a Database Connection).
-mse <hostname>	localhost	Dir	Sets the Media Sequencer to be used. If no parameter is defined it will try to use a local Media Sequencer.
-showcommands	<none>	Dir	Displays the Director Commands window.

8.1.2   Media Sequencer Configuration

The Media Sequencer Configuration window is used to enable the Director to connect to other devices; for example, a newsroom integration with connections to Viz Gateway, a connection to the Viz Pilot database, a GPI configuration of a GPI card on the Media Sequencer, etc.

To open the Media Sequencer Configuration window in Director, go to **Main menu > Tools > Media Sequencer Configuration.**

This section contains the following topics:

- Database Tab
- Viz One Tab
- MOS Tab
- General Purpose Input (GPI Tab)
- Video Disk Control Protocol (VDCP Tab)

Database Tab

Main menu > Tools > Media Sequencer Configuration > Database

The Database tab sets the database connection for the Media Sequencer. If it was added during Viz Pilot System Installation, the information is made available in this window. However, it must still be manually applied.

Media Sequencer Configuration
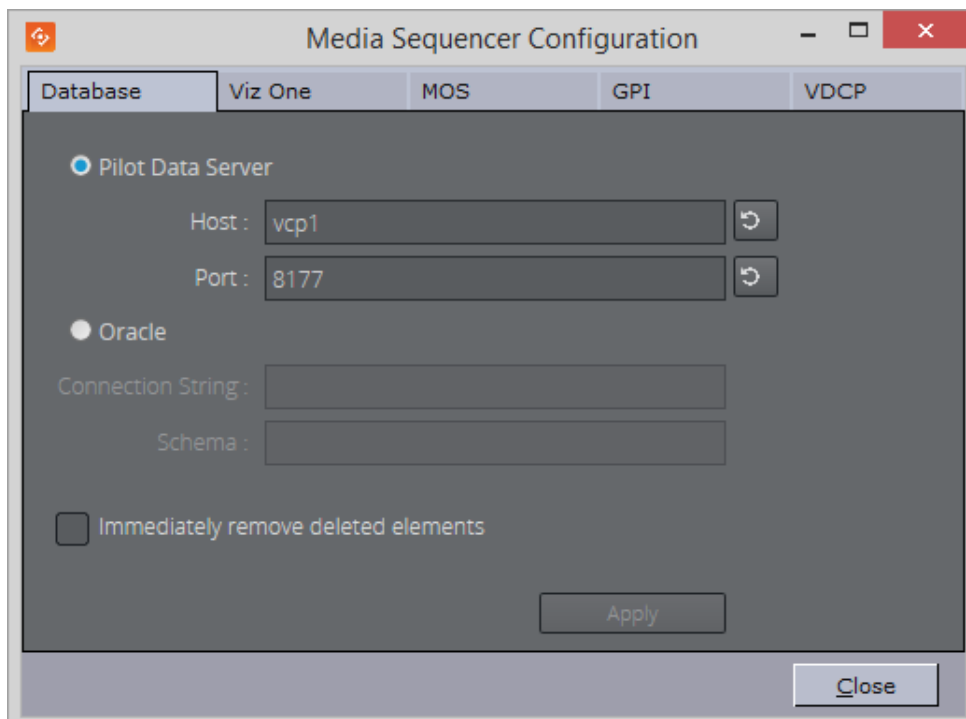
In Media Sequencer 4.0 or later, the Database tab allows you to configure whether Media Sequencer will fetch data elements directly from an Oracle database or via a Pilot Data Server.

> ⚠️ **Note:**
> - If Director is connected to a Graphic Hub, Media Sequencer must connect to a Pilot Data Server that is connected to the same Graphic Hub as Director.
> - If Director is connected to an Oracle database, Media Sequencer must connect to the same schema on the Oracle database, or connect to a Pilot Data Server that is connected to the same schema.

Properties And Parameters

- **Pilot Data Server:** Connect to the database through a Pilot Data Server. Enter the hostname and port of the Pilot Data Server.
- **Oracle:** Connect to an Oracle database directly.
    - **Connection String:** The connection string for Oracle, for example `pilot/pilot@<hostname>/<SID>`.
    - **Schema:** The name of the Oracle schema, for example `PILOT`.
- **Immediately remove deleted elements:** Elements that are deleted from the database will be removed from Media Sequencer.

Applying the Viz Pilot Database Settings

1. Select the **Media Sequencer Configuration** option from the **Tools** menu.
2. Select the **Database** tab, and enter the database settings.
3. Click **Apply**.

Viz One Tab

Main menu > Tools > Media Sequencer Configuration > Viz One
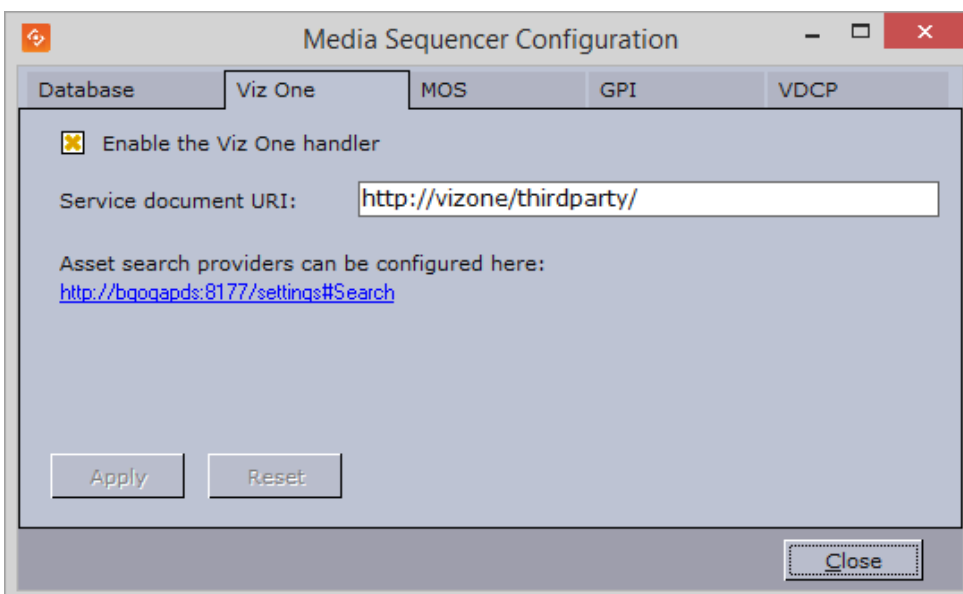
The Viz One tab sets the Viz One connection for Media Sequencer. The Viz One connection enables Media Sequencer to request transfer of media to Viz Engine for playout.

> ⚠️ **Note:** Director and Viz Pilot News get their Viz One connection parameters from the Viz Pilot database.

Properties and Parameters

- **Enable the Viz One handler:** Enables Media Sequencer to handle Viz One transfer requests.
- **Service document URI:** Sets the service document URI which describes the services the search will provide.

Applying Viz One settings

1. Select the **Media Sequencer Configuration** option from the **Tools** menu.
2. Select the **Viz One** tab.
3. Click the asset search provider link.
4. From the web page (under Search Providers) copy the relevant asset search provider URL and paste it into the **Service document URI** field in the Viz One tab.
5. Click **Apply**.

Working with Viz One

Viz Pilot can connect to media providers and repositories that support the Pilot Data Server REST API, like Viz One.

Connecting To Viz One

1. Install and configure your media provider (for example Viz One).
2. Install and configure your Pilot Data Server by configuring Viz One as a search provider (see **To configure search providers** (Viz One)). See the Pilot Data Server documentation.
3. Configure Director's Viz One Tab connection in order for Media Sequencer to request transfer of video files from Viz One to your configured Viz Engine for playout.
4. Configure your Viz Engine to enable transfer of video files over FTP from Viz One to Viz Engine for playout.
5. Configure your Director's output channels in order to playout video clips from the Viz Engine storage points where Viz One files are sent for playout (see Profile Configuration).
   If you're using Viz Engine to preview video clips from Viz One locally, you must also install Video Codecs (for example MPEG-4 codec and Haali Media Splitter).

MOS Tab

Main menu > Tools > Media Sequencer Configuration > MOS
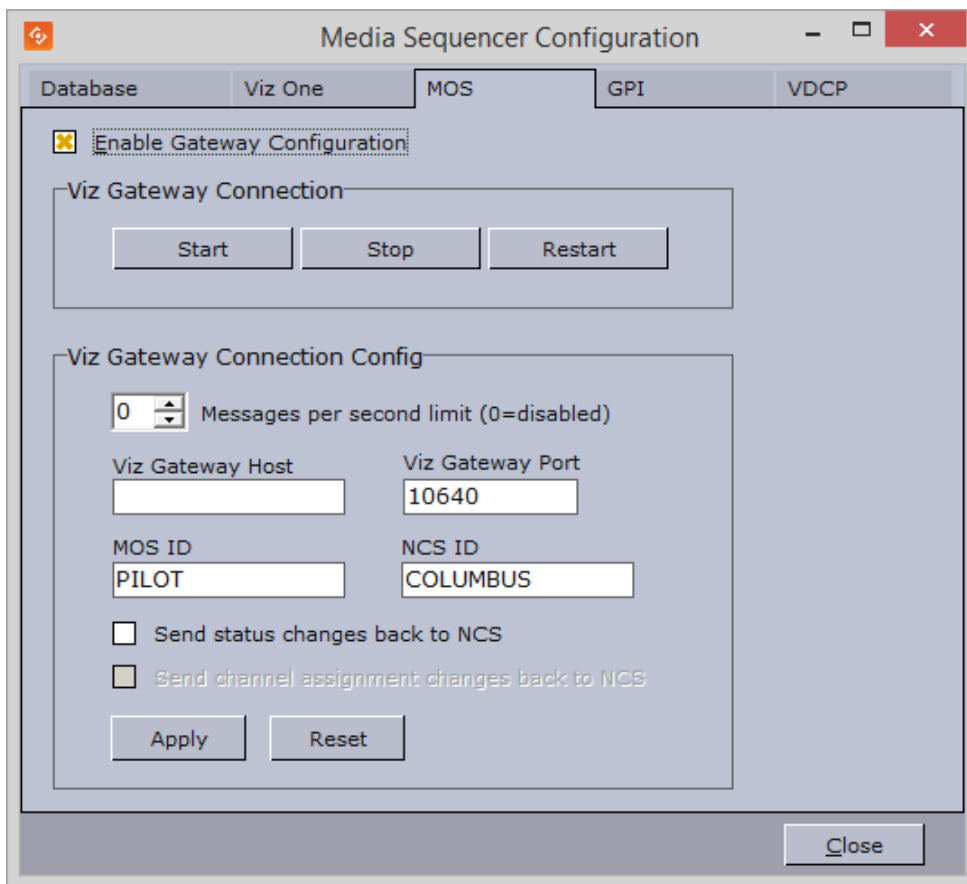
The MOS tab enables Media Sequencer to connect to Viz Gateway. If configured to use Viz Gateway 1.0, the newsroom system and Viz Gateway must be pre-configured with an NCS ID and MOS ID. For connections to Viz Gateway versions 2.0 and later, only the Viz Gateway host and port number are required.

> ⚠ **Note:** MOS IDs are configured on the NCS server, and are usually case sensitive.

Properties and Parameters

- **Enable Gateway Configuration:** When selected, enables the user to configure the Viz Gateway connection.
- **Viz Gateway Connection:** The controls in the Viz Gateway Connection section allows a start, stop, and restart action to be performed on a Viz Gateway connection.
    - **Start:** Starts a connection to Viz Gateway.
    - **Stop:** Closes a connection to Viz Gateway.
    - **Restart:** Restarts a connection to Viz Gateway.
- **Viz Gateway Connection Config:** Define Viz Gateway connection settings.
    - **Messages per second limit (0=disabled):** Sets how many messages per second should be sent from Viz Gateway to the NCS. This is done to prevent flooding of the NCS. 0 disables the message throttling allowing unlimited messages per second.
    - **Viz Gateway Host:** IP address of the Viz Gateway host.
    - **Viz Gateway Port:** Connection port for the Viz Gateway.
    - **MOS ID:** ID of the connecting MOS device.

- **NCS ID:** ID of the Newsroom Control System (NCS).
- **Send status changes back to NCS:** Sends new status changes to the NCS.
- **Send channel assignment changes back to NCS:** Sends changes to the NCS.

❌ **IMPORTANT!** A database connection must be established in order for the MOS integration to work. See how Media Sequencer Configuration#To apply the Viz Pilot Database settings for details.

Configuring a Viz Gateway Connection

1. Start Director.
2. Select **Profile Configuration** from the **Tools** menu (see Profile Configuration).
3. Click the **MOS** tab, and enter the appropriate settings collected from the newsroom system and the Viz Gateway host.
4. Click **OK** and check that the status indicator (G) appears in the lower right of Director's status bar.
5. Click the Playlists button in the Resource Panel, and check that MOS Playlists are added to the list of available playlists.

General Purpose Input (GPI Tab)

Main menu > Tools > Media Sequencer Configuration > GPI
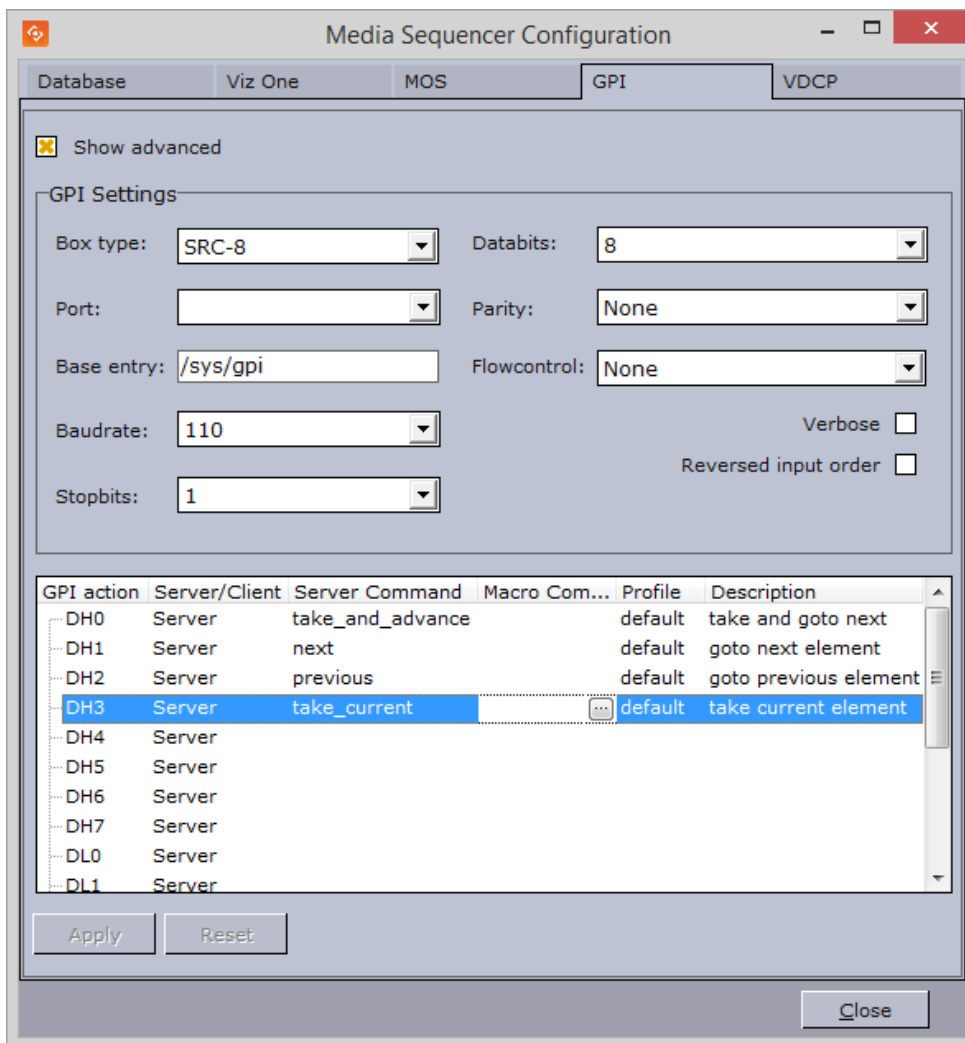
The GPI tab allows Media Sequencer to be configured to handle GPI commands. The commands can be handled by Media Sequencer itself (Server Command) or forwarded to Director (Macro Command).

Both server and client side commands are profile specific, meaning that the profile determines which Director client and potentially which Viz Engine(s) execute a client and/or server command. It is therefore very important to assign the correct profile for each GPI action as they refer to unique profiles configured per Director client.

For example, if a Director has its own profile with two Viz Engines (program and preview) and receives a client command from a GPI on Media Sequencer, the command executes some logic on Director that issues commands to the program renderer. If the correct profile is not set, such commands might end up on the wrong Director and potentially on the wrong program renderer. The same would happen if the GPI action was defined as a server command; however, it would then trigger commands from the Media Sequencer to the Viz directly and not through Director.

This section contains the following topics:

- Properties and Parameters
- Assign a Server Command
- Assign a Client Command
- Add a Macro Command

Properties and Parameters

- **Show advanced:** Displays all the settings in the GPI tab.
- **Box Type:** Select the type of GPI box that is being configured. The Box Type can be set to SRC-8, SRC-8 III or SeaLevel.
- **Port:** Sets the port that the GPI box is connected to. The Port can be set to COM1-COM17, or None.
- **Base Entry:** This is the node in the Media Sequencer's data structure where the systems look for the GPI actions. The base entry is by default set to /sys/gpi.
- **Baudrate:** Sets the maximum rate of bits per second (bps) that you want data to be transmitted through this port. The Baudrate can be set to 110-921600. It is recommended to use the highest rate that is supported by the computer or device that is being used.
- **Stopbits:** Sets the interval (bps) for when characters should be transmitted. Stopbits can be set to 1, 1.5, or 2.
- **Databits:** Sets the number of databits that should be used for each transmitted and received character. The communicating computer or device must have the same setting. The number of databits can be set to 5, 6, 7 or 8.
- **Parity:** Changes the type of error checking that is used for the selected port. The communicating computer or device must have the same setting. The parity can be set to:
  - **Even:** A parity bit may be added to make the number of 1's in the data bits even. This will enable error checking.
  - **Odd:** A parity bit may be added to make the number of 1's in the data bits odd. This will enable error checking.
  - **None:** No parity bit will be added to the data bits sent from this port. This will disable error checking.
  - **Mark:** A parity bit set to 0 will be added.
  - **Space:** A parity bit set to 1 will be added.
- **Flowcontrol:** Changes how the flow of data is controlled. The Flowcontrol can be set to:
  - **None:** No control of dataflow.
  - **XonXoff:** Standard method of controlling the flow of data between two modems. XonXoff flowcontrol is also known as software handshaking.
  - **Hardware:** Standard method of controlling the flow of data between a computer and a serial device. Hardware flowcontrol is also known as hardware handshaking.
- **Verbose:** If enabled, Media Sequencer's GPI handler outputs log information. This information is useful for debugging.
- **Reversed Input Order:** Note that this check box is only available if Box Type is set to SRC-8. If enabled, the signal line that originally triggered GPI action DL0/DH0 will now trigger GPI action DL7/DH7, the signal line that originally triggered GPI action DL1/DH1 will now trigger GPI action DL6/DH6, and so on.
  Commands and actions list:

- **GPI action:** Shows a list of the available GPI actions.
- **Server/Client:** Shows a drop-down list box in every row. This determines whether the selected GPI action applies to Media Sequencer (Server option) or the local Director (Client option).

> ⚠ **Note:** The server and client actions are mutually exclusive.

> ⚠ **Note:** GPI actions for the Client option will only work if Director is running.

- **Server Command:** Shows a drop-down list box in every row, where the action to be performed on this GPI line can be selected. Server commands are GPI actions that apply to the Media Sequencer. When right-clicking an item in the playlist, the **Playlist** menu opens. In this menu, select **Cursors**, and then **Set Next**. An arrow appears next to the selected element in the playlist, which indicates that this is the current GPI cursor. The server commands can be set to:
    - **take_and_advance:** Runs the Start operation on the current element, and then shifts to the next element in the playlist.
    - **take_current:** Runs the Start operation on the current element (the element with the cursor).
    - **next:** Shifts to the next element in the playlist.
    - **previous:** Shifts to the previous element in the playlist.
    - **continue:** Runs the Continue operation on the current element.
    - **out_current:** Runs the Take Out operation on the current element.
- **Macro Command:** Macro commands are silent GPI actions. Clicking the ellipsis (**...**) button opens the Add Command window.
- **Profile:** Sets the profile to be used for the GPI action. This profile must match the profile set for the playlist that is to be triggered by the GPI actions. The drop-down list shows the profiles configured on Media Sequencer.
- **Description:** Shows the description of the GPI action, as it was specified in the **Add Command** window.

Assigning a Server Command

1. Select a GPI action, and select **Server** in the Server/Client column.
2. Select a **Server Command**.
3. Select a **Profile**.
4. Click **Apply**.

Assigning a Client Command

1. Select a GPI action, and select **Client** in the Server/Client column.
2. Select or create a **Macro Command**.
3. Select a **Profile**.
4. Click **Apply**.

Adding a Macro Command
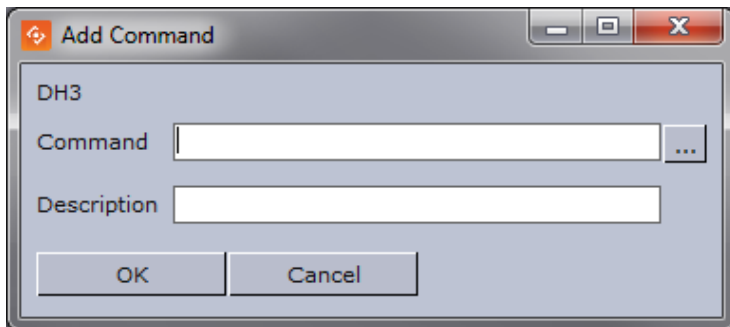
1. Select the **Macro Command** column, and click the small ellipse (**...**) button to open the **Add Command** dialog box.
2. Enter the command in the **Command** text field, or alternatively click the ellipse (**...**) button to open the Predefined Functions window.

Video Disk Control Protocol (VDCP Tab)

Main menu > Tools > Media Sequencer Configuration > VDCP

Using Video Disk Control Protocol (VDCP), the Media Sequencer acts like a server that controls the graphics through the VDCP protocol. The Media Sequencer sets up a connection with a video controller, over which VDCP commands are sent; in this way the video controller is able to communicate with the playlist or show.
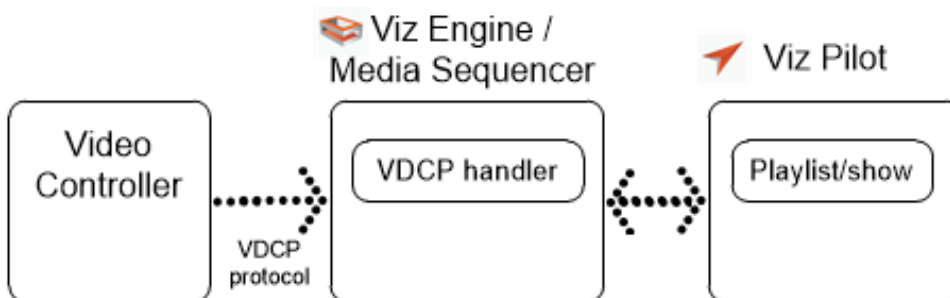
> ⚠ **Note:** The VDCP protocol defines recommended serial settings. For configuration of other settings, see **Electrical and Mechanical Specifications** in the Media Sequencer manual's VDCP section.

This section contains the following topics:

- VDCP Tab
- VDCP Properties and Parameters
- VDCP Macros

VDCP Tab

The VDCP tab always shows an up-to-date status, where changes made by others are shown live. All changes are immediately propagated to Media Sequencer.

To **open** the VDCP Tab, either:

- Go to the main menu: Tools > Media Sequencer Configuration > VDCP tab
- Select Configure VDCP from the playlist's context menu in Director ***
- Once VDCP handlers have been defined, click the VDCP Indicator button at the bottom of each playlist ***
  - Not available if using the Legacy Playlist style.
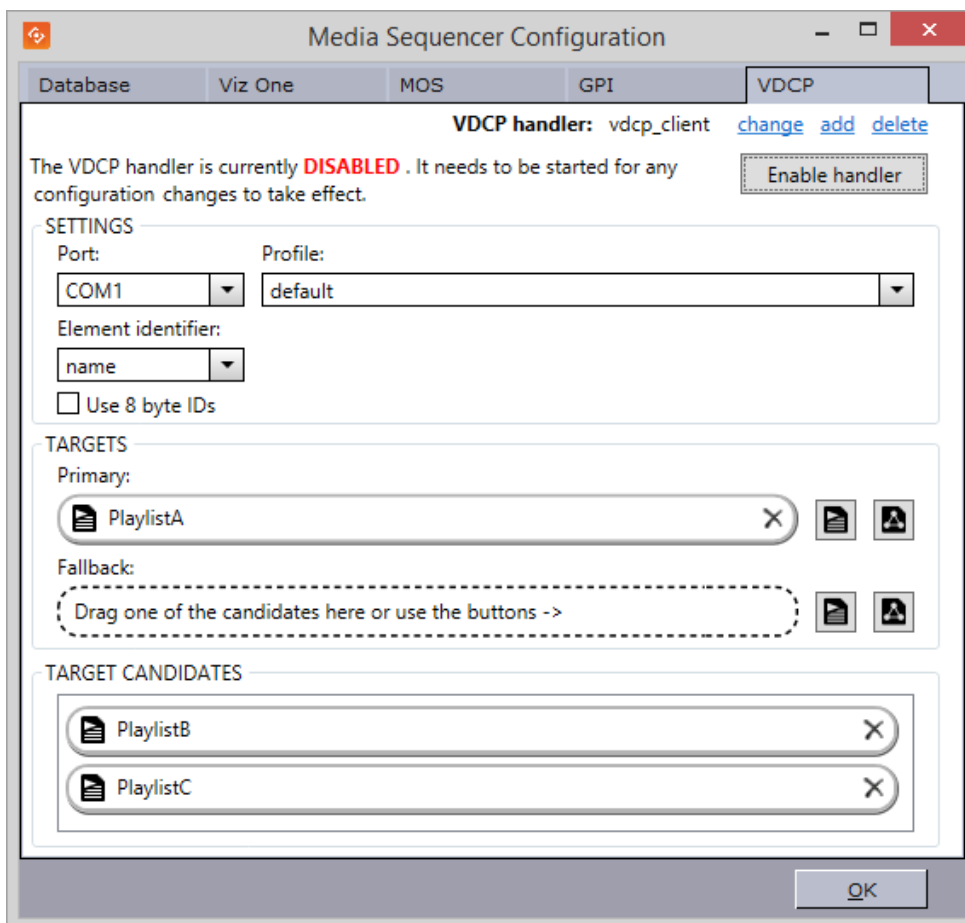
VDCP Properties and Parameters

The configuration consists of selecting general VDCP and playlist control settings.

- **VDCP handler:** This displays the currently selected VDCP handler. You can **change**, **add** or **delete** multiple VDCP handlers.
- **Enable/Disable handler:** The handlers can only be modified when they are disabled.
- **Port:** Select the appropriate COM port for the selected VDCP handler.

- **Profile:** Select the profile to use. The profile defines the Viz Engines and video devices where the command is to be executed. In the VDCP handler, the configured playlist must be active on the configured profile. (See also Profile Configuration).
- **Element identifier:** Use either the element's **name** or **description** as the VDCP ID. The Name is a unique identifier, assigned to each element by the Media Sequencer, whereas the Description is the textual description of the data element, as entered when the item was created in Director/Viz Trio.
- **Use 8 byte IDs:** Allow the use of 8 character IDs. For more information about 8 byte IDs and the Media Sequencer VDCP plugin see the **8 characters identifiers** section in the **Media Sequencer User Guide**.
- **Targets:** Select the Director playlists or Viz Trio shows which will be the Primary and Fallback targets. When the vdcp_client handler receives a command over the VDCP protocol, it iterates over the **primary target** collection, looking for the element with the specified ID. If a matching item is found, the command is executed on it. If no match is found in the primary target collection, or if there is no primary target collection specified, then it searches the **fallback target** collection. If there are still no matches, an error is returned over the VDCP protocol, and it is also reported in the Media Sequencer logs.
  Click the buttons next to the primary and fallback target slots to select a playlist or show. Double click a slot to clear it.
  A 'Path not found' warning, or a missing playlist/show icon, indicates that the selected playlist or show is invalid (for example, if the playlist has been deleted).
- **Target Candidates:** Suggests some of the playlists and shows that are available. If you do not find the one you want, use the buttons next to the Primary and Fallback target slots.
  Drag a candidate to a target slot or double-click one to make it the Primary target.

VDCP Macros

The following macros provide easy access for turning the VDCP control on and off for an open playlist. They only work with the VDCP handler called *vdcp_client*. Note that using the VDCP macros will restart the VDCP handler.

```
vdcp:clear_all_targets, vdcp:clear_primary_target, vdcp:clear_fallback_target,
vdcp:set_playlist_as_primary_target, vdcp:set_playlist_as_fallback_target,
vdcp:set_primary_target, vdcp:set_fallback_target
```

8.1.3   Configuring the Pilot Edge Window

For Viz Pilot Edge versions 1.6 and later, templates created or edited in Template Builder 1.3 or later open in a Viz Pilot Edge window inside Director.
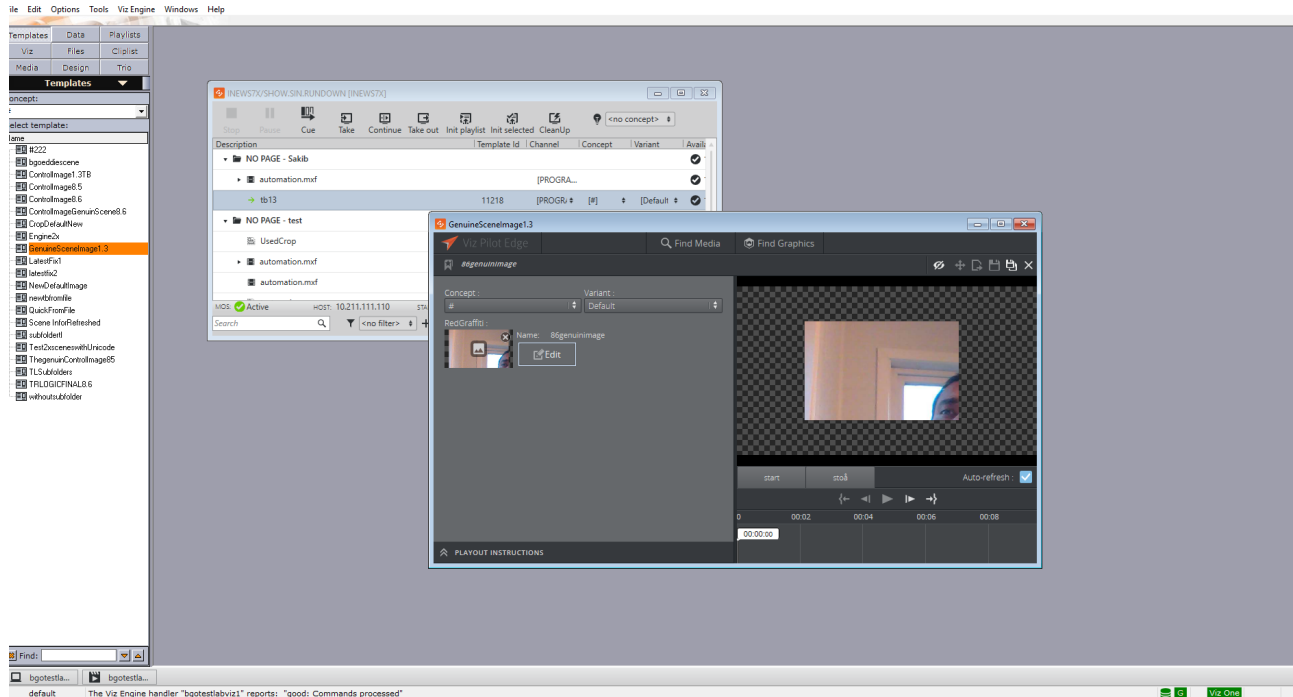
> ⚠️ **Note:** This feature must be configured in Pilot Data Server 8.5 or later, see the *Pilot Data Server Administrator Guide* for more details.

8.1.4   Working with Stillstore Images
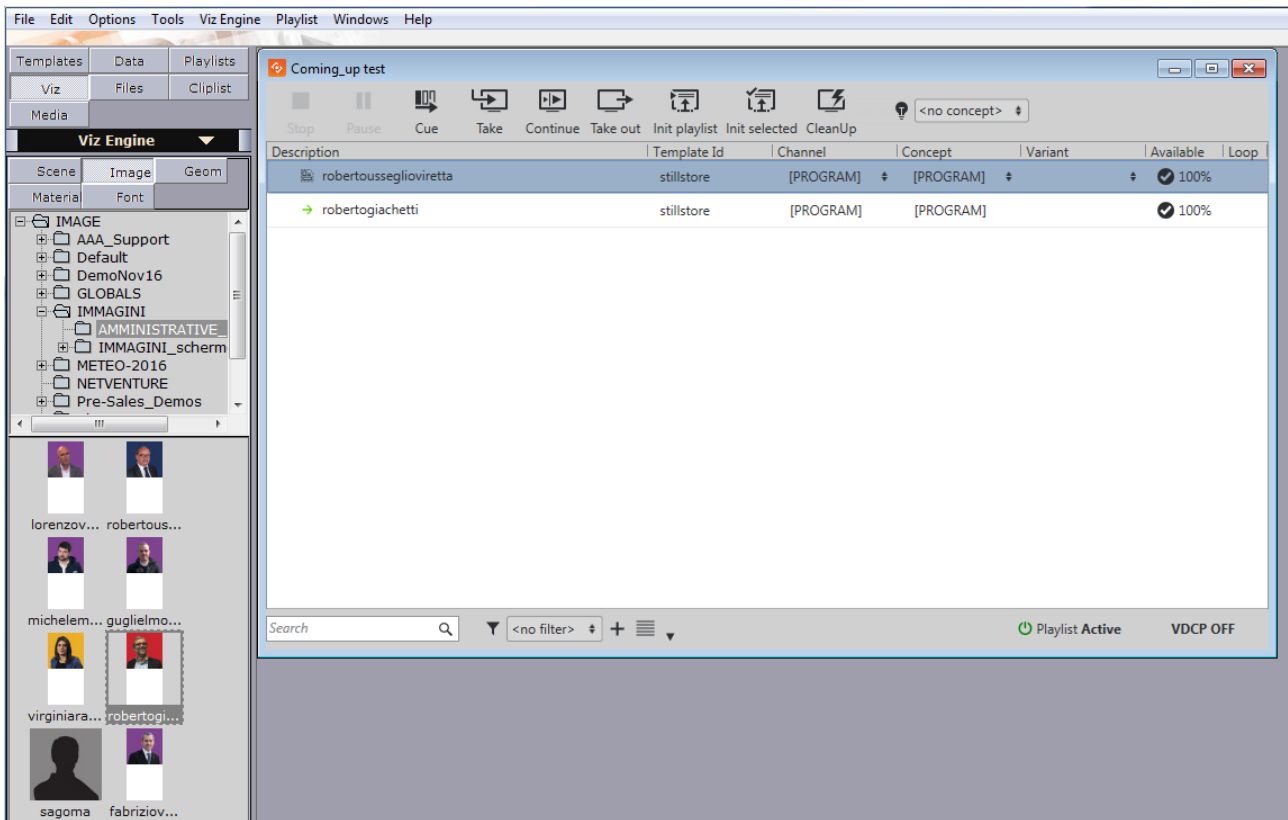
When searching for still images in Director, whether from Object Store, Viz One, Viz Engine or only from the file system, Director lets you drag and drop any still image into the playlist for playout without the need for a graphics template. However, current limitations with Director's Graphic Hub integration require a default stillstore scene to be present in order to enable playout of fullscreen stillstore images.

> ⚠️ **Note:** The stillstore scene must be named *vizrt_stillstore*, and must be placed in Graphic Hub's *data* (root) directory.

This section explains how to work with stillstore scenes.

Creating a Stillstore Scene in Viz Artist

Vizrt provides a basic stillstore scene which can be imported and used. However, in order to get the look and feel required to match the overall theme, it is recommended to further improve the scene.

In addition, it is recommended to use the Object Store functionality to add keywords to images that are used for fullscreen playout. Fullscreen images also have a fixed ratio to comply with the final design and desired output.

The scene example below describes the required elements and parameters for the stillstore scene.

Creating a Stillstore Scene
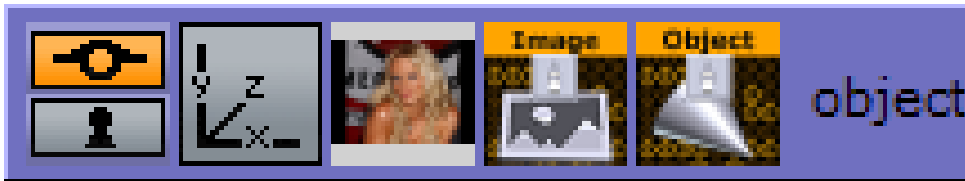
1. Start **Viz Artist**.
2. Add a **Group** container to the scene tree.
3. Click **Built Ins__**on the main menu, and then press the **CTRL + 2** keys to open the **Container Plugins** view.
4. Select the **Control** folder.
5. Add the **ControlImage** plugin to the **group** container.
    - Set ControlImage's **Field identifier** attribute to **1**.
    - Adding a control plugin automatically adds the **ControlObject** plugin to the **group** container.
6. Add an image to the container.
7. Rename the **group** container to **object** (lower case).
8. *Optional*: Design your scene.
9. Save the scene as **vizrt_stillstore**.

Moving the Stillstore Scene to the Graphic Hub Root

Graphic Hub is the graphics database where all scenes created in Viz Artist 3.x are stored and fetched for playout on Viz Engine. Graphic Hub has a tool named Graphic Hub Manager (GHM) that is used to manage the databases.
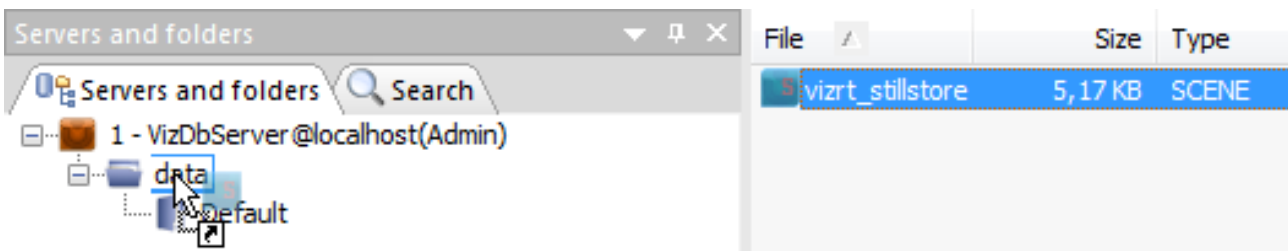
1. Start Graphic Hub Manager.
2. Login as **admin**.
    - The standard out-of-the-box password for administrators is **VizDb**.
3. Open the project or folder containing the **vizrt_stillstore** scene.
4. Select the scene, press **CTRL** and drag and drop it onto Graphic Hub's **data** (root) project folder.
    - Alternatively, copy and paste the scene using the keyboard keys or context menus.

> ⚠ **Note:** Director's stillstore scene is located in the *scenes_to_be_dragged_to_root* project folder.

8.1.5   Profile Configuration
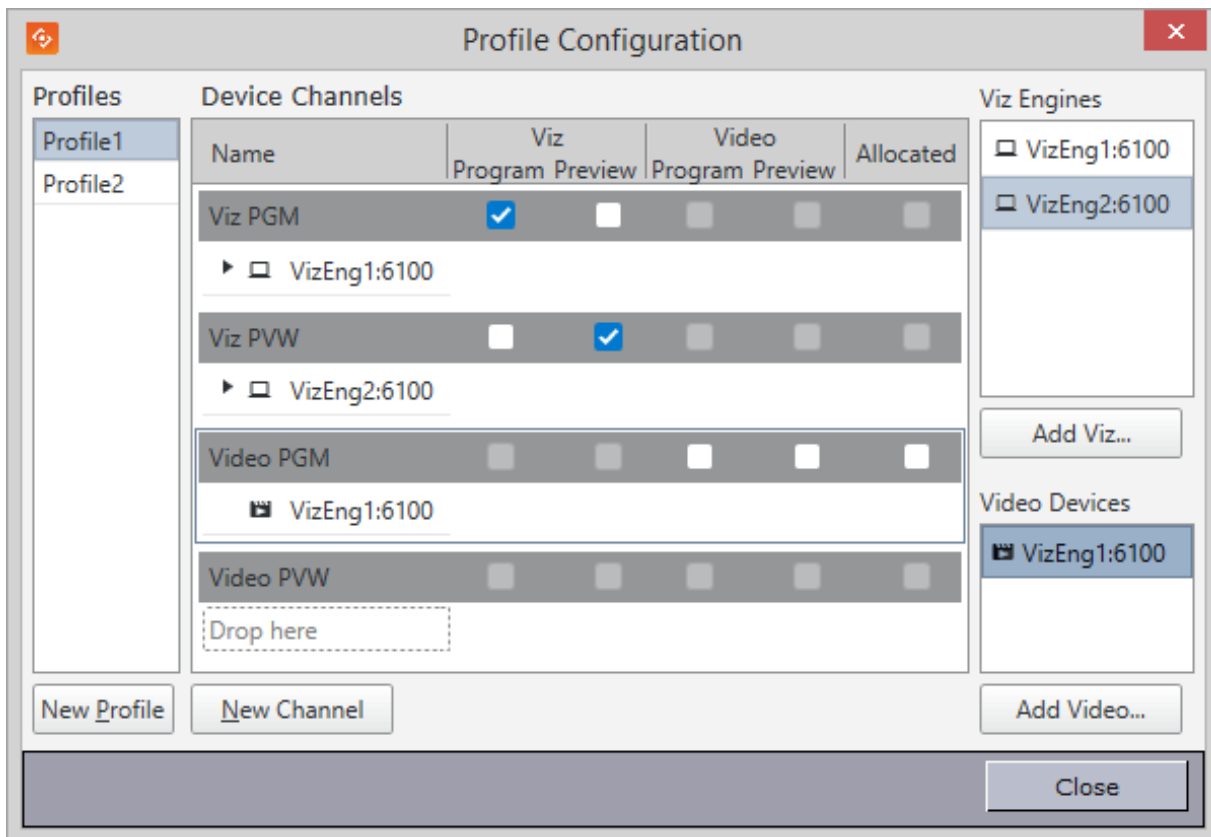
An important part of the profile configuration is to create profiles for different purposes. For example, Profile Setups can be created where different channels have the program and preview function. This makes it easy to switch between different output settings.

This section contains the following topics:

- Categories and Channels
- Profile Setups
- Forked Execution
- Configuring Profiles
- Configuring Channels
- Configuring Devices

Categories and Channels

Setting the Default Channel for a Template

1. In Template Wizard, add Categories and Channels.
2. Map the new category to the desired channel.
3. In Template Manager, edit a template variant and select the required Category from the drop-down list. See Working with Variants.

4. In Director, create a data element from the template and drag it to the playlist.
   - The default channel of the item is the channel selected in step 2.

✅ **Tip:** The default channel can also be passed on to a newsroom system if required. See Get the data element's default channel.

⚠️ **Note:** When setting categories and channels to filter playlists, the channel configuration that is configured in Template Manager must correspond with the actual channel name.

⚠️ **Note:** Any Legacy template that does not use Control Object is unable to assign itself to the correct program channel. It will therefore always default to the default program channel in the profile configuration, and disregard information in the Categories and Channels setup.

Profile Setups

Profile	Program	Preview
Main	A	B
Backup	B	

A typical example of when it makes sense to use different profiles is in the case of a backup configuration where a switch from a main to a backup renderer is required. For example, if two output renderers are named A and B, where A is program and B is preview, a profile named "Main" will then have channel A as program and channel B as preview.

If the renderer acting as program (A) in the "Main" profile fails, a profile named "Backup" could have channel B as program.

⚠️ **Note:** a channel can be designated as a Program or Preview channel by selecting the check box in the appropriate columns (Program or Preview); however, the program and preview channels are reciprocally exclusive - only one channel can be set to program, and only one channel can be set to preview. For example, if A is set to program and B is set to preview, and program is then changed to C, A will no longer be set as program.

GPI

Another example of a typical use of profiles is when playout is controlled through a General Purpose Input (GPI), for instance through hardware such as a vision mixer.

When GPI is enabled, the external cursor (the GPI system's cursor) will be shown in any client that is using the same profile as the external system. A typical setup would be that one Viz Trio client is in the same profile as the GPI system, and functions as a prepare station for the producer sitting at the vision mixer desk. Data elements are then prepared and displayed on a preview visible to the producer, before being triggered from the vision mixer. This configuration needs a separate "GPI" profile that is not used by other control application clients. Other clients can be in other profiles and produce content to the same output channels. However, they must be on other transition layers or on another Viz layer so that they will not interfere with the graphics controlled by the external system.

VDCP

If you are browsing the Cliplist for clips on other video devices (that support the Multiport Video Control Protocol - MVCP) you may add these clips for manual playout in any playlist regardless of the profile configuration.

Forked Execution

This section describes how forked execution can be used with Standalone scenes and Transition logic scenes.

Forked Execution replaces the earlier method of rendering specific scene container by name (see **Visible containers** below).

Standalone Scenes

Forked execution supports standalone scenes by executing the same graphics with different concepts on two or more render engines. Concepts are defined per channel when working with the Changing Settings.

As an option you can also use this setup to handle fail situations by having the same graphics concept rendered on both engines.

Transition Logic Scenes

As with Standalone scenes, forked execution supports setting concepts for Transition Logic scenes. In addition, by defining channels with different render engine setups Transition Logic scenes can show different states of the same scene on a per engine basis. All states are synchronized for all engines (at all times) in order to achieve an artifact-free and smooth morphing of the graphics from one state to the other.

For example, if you have three render engines you can set up a range of channels with different combinations of render engines per channel.

Channel and Engine Combinations

Channel	Viz Engines
A	1,2,3
B	1,3
C	1
D	2,3
E	2
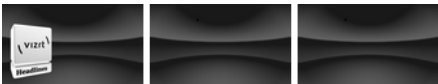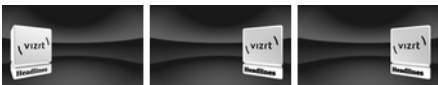
For example, if you have a scene with four layers, each layer can be controlled separately from the other layers (see table below). By setting a state per layer you can achieve a varied output depending on the channel used and how that channel is configured in terms of render engines (see table above).
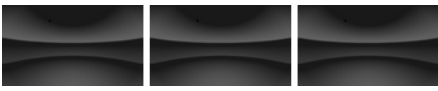
- Layer 1: Shows and hides a geometry (for example a cube)
- Layer 2: Shows and hides a geometry (for example a cube)

- Layer 3: Positions the geometries.
- Layer 4. Animates the layer 1 geometry by showing the next image or a logo.

States for layer 1	States for layer 2	States for layer 3	States for layer 4
Show cube	Show text	Position left	Next image
Hide cube	Hide text	Position right	Show logo
<ignore>	<ignore>	Position center	<ignore>
		<ignore>	

With the scene layers and configured channels listed above you can have the following output on each of the three engines:

Channel	Layer state	Output on Viz Engine 1, 2 and 3
Channel C(Viz 1)	Show cubePos leftShow textShow logo	
Channel D(Viz 2,3)	Show cubePos rightShow TextShow Logo	
Channel E(Viz 2)	Pos centerHide text	
Channel A(Viz 1,2,3)	Next image	
Channel A(Viz 1,2,3)	Next image	
Channel A(Viz 1,2,3)	Show textNext image	
Channel A(Viz 1,2,3)	Show logo	
Channel B(Viz 1,3)	Hide cube	

Channel	Layer state	Output on Viz Engine 1, 2 and 3
Channel E(Viz 2)	Hide cube	

Visible Containers

Director still supports (although it is deprecated), a behavior similar to that of forked execution. By designing a standalone scene where each root container is a variant of the other, you can configure each render engine to render specific scene containers by name. In effect, one or more containers can be assigned and rendered visible by a single render engine for a scene with two or more root containers.

> ⚠️ **Note:** Due to potential performance issues when using large textures (for example HD) this option is no longer recommended. Concept and variant design of standalone and transition logic scenes is therefore the recommended design convention.

Configuring Profiles

Opening the Profile Configuration

1. Select **Tools** from the main menu, and then **Profile Configuration (CTRL+M)**, or
2. Right-click the profile on the status bar (lower-left) and select **Profile Configuration...** from the context menu.

Adding a New Profile
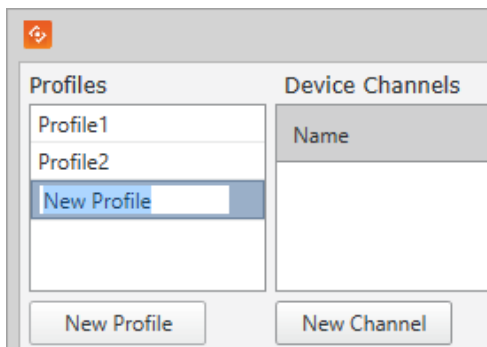
In the Profile Configuration window click the **New Profile** button, enter a new unique profile name, then press **Enter**.

Renaming a Profile

1. Right-click the profile and select **Edit Profile Name**, or double-click it and enter the new name.
2. When finished editing the name, press **Enter** or click the cursor outside the Profiles list.

Deleting a Profile

- Right-click the profile and select **Delete Profile**, or select it and press the **Delete** button.

> ⚠ **Note:** Media Sequencer requires a profile to be present in the system. If all profiles are deleted, an empty vizroom1 profile is automatically added.

Setting a Profile Password

Password protected profiles display a lock icon next to them in the profiles list. A password is required to be able to select these profiles.

Create Or Change A Profile Password

1. Right-click the profile and select **Set Password For Profile**.
2. Type the password in the **Change Password** dialog.

Sorting the Profiles List

1. Right-click the profiles list or a profile name and select **Sort**/**Sort List**.
2. Choose one of the following sorting options:

- **Unsorted:** The list retains its original order.
- **Ascending:** The profiles are sorted alphabetically in an ascending order.
- **Descending:** The profiles are sorted alphabetically in a descending order.

The chosen profile sorting option is remembered upon restart.

Configuring Channels

This section contains the following topics:

- Add an Output Channel to the Channels List
- Add a Concept Override for a Channel's Output Device
- Rename an Output Channel in the Channels List
- Remove an Output Channel from the Channels List
- Remove an Output from the Channels List

Adding an Output Channel to the Channels List
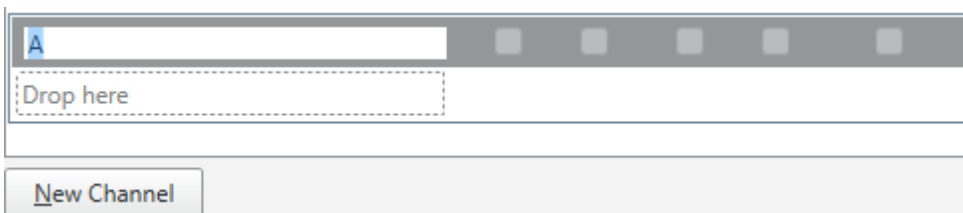
Click the **New Channel** button, or drag and drop a Viz Engine or video device to the Channels list.

Adding a Concept Override for a Channel's Output Device
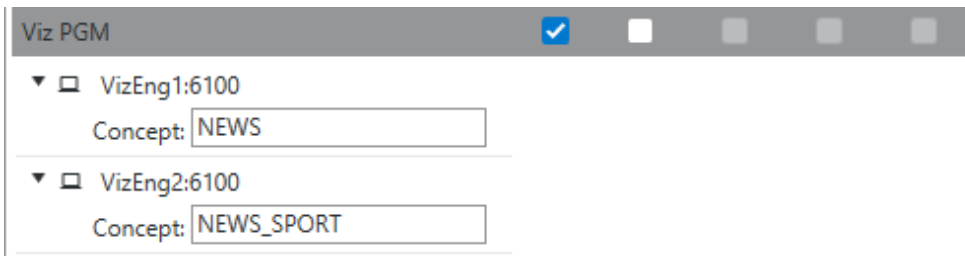
1. **Expand** the channel's output device and append the concept name.
   This will override any concepts set elsewhere.

> **ⓘ Info:** Concept names are case sensitive.

2. Click **OK**.

Renaming an Output Channel in the Channels List

Right-click the channel and select **Edit Channel Name** from the context menu, or double-click the name.

Removing an Output Channel from the Channels List
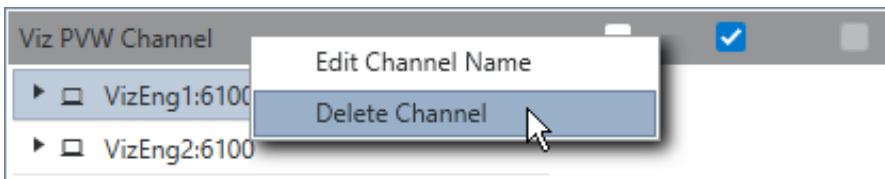
Right-click the channel entry and select **Delete Channel** from the context menu, or select the channel and press the **Delete** button.

Removing an Output from the Channels List

Right-click Viz Engine or video device and select **Delete Output**, or simply select it and press **Delete**.

Allocating Video Clips

Checking the **Allocated** check-box allocates video devices for the clips in the playlist. Media Sequencer enables the transfer of all clips to all video devices for two channels with each their video device. Based on the allocation, the

clips in the playlist are played out in a given order (for example A, B, A, B) to allow back-to-back playout. Allocation is not necessary for video devices that are able to play out back-to-back.

Configuring Devices

This section contains the following topics:

- Add a Viz Engine
- Add a Video Device
- Edit a Viz Engine or video device
- Delete a Viz Engine or video device
- Enable scene transitions
- Enable still preview
- Render specific scene containers
- Set a different font encoding

Adding a Viz Engine

1. Click the **Add Viz**... button in the Profile Configuration window to open the Configure Viz Engine dialog box.
2. Enter the **Hostname** and **Port** (default port is 6100).
3. *Optional*: Select **Mode**
   - **Scene Transitions:** See how To enable scene transitions.
   - **Still Preview:** See how To enable scene transitions.
4. *Optional*: Select a Viz One storage in **Asset Storage**.
5. *Optional*: In Deprecated Settings, select Visible containers and/or Font Encoding.
6. Click **OK**.

A status indicator will show if the renderer is on-air.

> ⚠ **Note:** The Asset Storage drop-down lists the available Viz Engine storage points where Viz One files can be sent for playout.

Adding a Video Device

1. Click the **Add Video**... button in the Profile Configuration window to open the Configure Video Device dialog box.
2. Select the video server **Type:**
   - Viz Engine
   - MVCP Compatible: Multiport Video Control Protocol, see Cliplist.
   - Dummy Server: A dummy server can be part of a channel used for timing GFX in Timeline Editor. Used when it is necessary to adjust timing based on lower clips.
3. Enter the **Host** or **IP** address and the **Port:**
   - Default port for MVCP is 5250.
   - Default port for Viz Engine is 6100.
4. Select **Asset Storage**. Select a publishing point from the list, so that clips or data elements are transferred to the right location (the specified Viz Engine) for playout.
5. Enable/Disable **Fullscreen** mode.
   - When Fullscreen is **enabled**, which is the default behavior, stand-alone video clip elements are played out full screen in the back layer. Graphics in the middle or front layer will still play.
   - When Fullscreen is **disabled** stand-alone video clip elements will only replace the current video clip without triggering animations or transition effects.
6. Select **Clip Channel** (only relevant for Viz Engine).

⚠️ **Note:** Viz Engine 3.6 and later supports playout on up to 16 clip channels. A clip channel might be unavailable for various reasons, for example, if it is configured to be inactive in Viz Config, or if the license only covers a limited number of clip channels.

7. Select whether or not to **Use Transitions**.
8. Select **Alternate Clip Channel**.

9. *Optional:* Enter the **Fullscreen Scene**.
10. Click **OK**.
     A status indicator shows if the video device is on-air or online.

Editing a Viz Engine or Video Device

Right-click the device and select **Edit** from the context menu, or **double-click** it.

Deleting a Viz Engine or Video Device

Right-click the device and select **Delete** from the context menu, or select it and press the **Delete button**.

Adding an Output to the Channels List

Drag and drop a Viz Engine or video device onto the channel in the Channels list, or select it and select **Add to profile** (creating a new channel) or **Add to selected channel**.

Enabling Scene Transitions

Transition Effect scenes are made in Viz Artist and can be applied to data elements to create custom transition effects from one scene to the other. If an effect is specified, the effect will be shown when the scene is taken on-air. Effects include wipes, dissolves, alpha fades, etc.
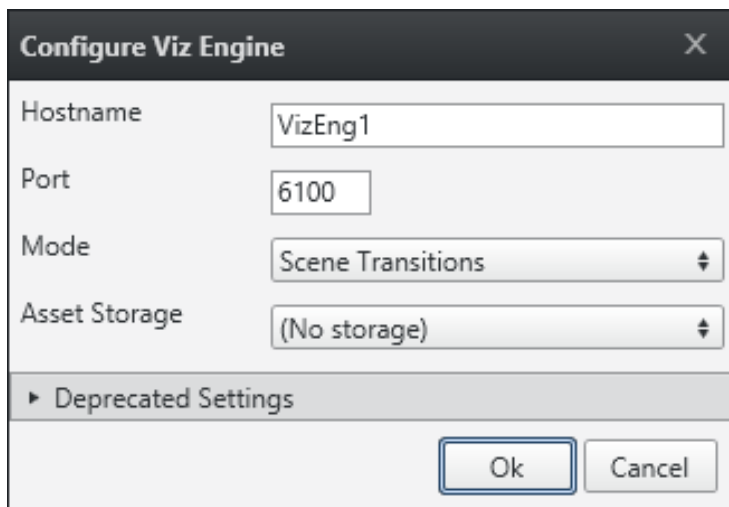
1. Configure the Viz Engine settings as seen in Adding a Viz Engine, above.
2. Set the **Mode** to **Scene Transitions** to allow the renderer to copy (or snapshot) the scenes to create a transition effect between them.
3. Click **OK**.
4. Add the program renderer to the program channel.

> ⚠ **Note:** In order to see the effects, the program channel must be configured and on-air.

Enabling Still Preview

See Configuring Remote Preview in Director.

Rendering Specific Scene Containers

Before configuring your Viz Engines to render specific scene containers< it is recommended to read the Forked Execution section for further explanation.

1. Configure the Viz Engine settings as seen in Adding a Viz Engine, above.
2. Click the **Deprecated Settings** button to expand the editor.
3. Click the **Add** button to enter the scene's container name that should be rendered visible.
4. *Optional*: Repeat step 3 to **Add** other containers.
5. Click **Ok**.

Changing Font Encoding

1. Configure the Viz Engine settings as seen in Adding a Viz Engine, above.
2. Click the **Deprecated Settings** button to expand the editor.
3. Select **Font Encoding**, it sets the font encoding of Viz Engine. The encoding can either be set to UTF-8 or ISO-8859-1. Default is UTF-8.
4. Click **Ok**.

8.2   Keyboard Configuration And Shortcuts

8.2.1   Keyboard Configuration
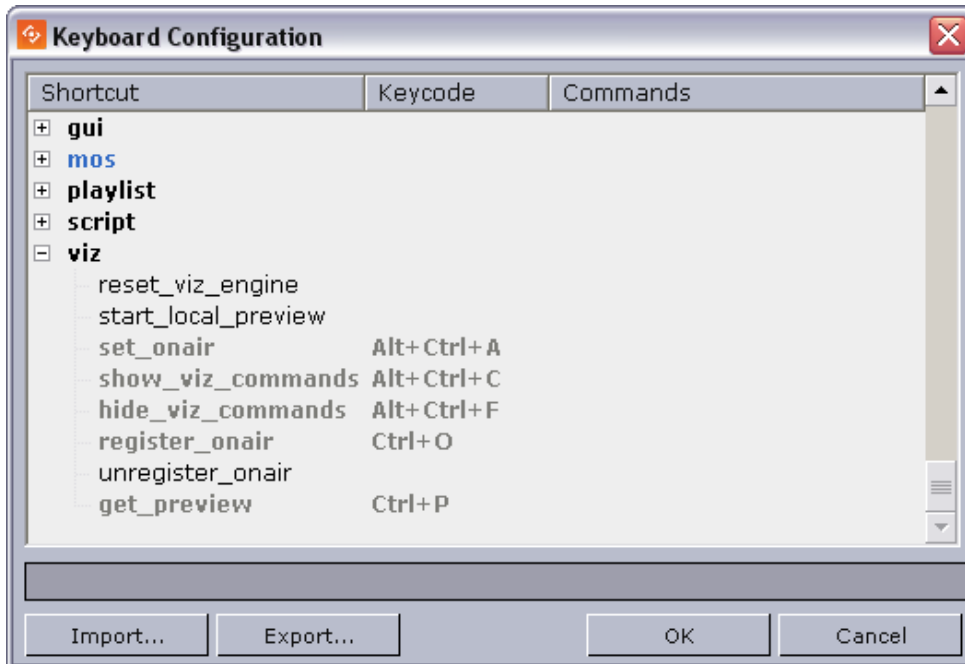
The Keyboard configuration window lets you define macros assigned to specific key combinations, pre-defined functions and VB Scripts. The latter is an option to further enhance and adapt the commands. It is also possible to configure specific key combinations to trigger GPI actions from Director.

To open this window, Select **Keyboard Configuration** from the **Options** menu. A selection of the configured commands is displayed. A full list of the Predefined Functions is available by selecting the **Add Macro** option in the context menu.

⚠ **Note:** The recommended alternative to GPI triggering using the LPT ports in Director is to add a GPI card to the Media Sequencer machine. This allows the user to trigger client side macros and pre-defined functions forwarded by the Media Sequencer to Director or server side commands to the Media Sequencer using its pre-defined commands.

Keyboard Configuration Window

The Keyboard Configuration columns:

- **Shortcut:** Displays the macro name.
- **Keycode:** Displays the keyboard shortcut.
- **Commands:** Displays the shortcut command.

The Keyboard Configuration buttons:

- **Import:** Opens the **Open** window where a previously exported KBD (keyboard) file can be imported.
- **Export:** Opens the **Save As** window where the current settings can be exported as a KBD file.
- **OK:** Saves the keyboard shortcut configurations, and closes the window.
  File location: `%PROGRAMDATA%\Vizrt\Viz Pilot x.y\Config\keyboard.kbd`
- **Cancel:** Closes the window.

The context menu contains the following options:

- **Add Macro:** Opens the Macro Shortcut window, where a new macro can be created.
- **Delete Macro:** Deletes the currently selected macro. Alternatively, click the keyboard shortcut **CTRL + DELETE**.
- **Remove Shortcut:** Deletes the shortcut key code of the currently selected macro. Alternatively, click the keyboard shortcut **DELETE**.

Working with Macros and Shortcuts

- Add a macro command: select **Add Macro...** .from the context menu to open the Macro Shortcut window, then enter the key combination, macro category:macro name and commands.
- Edit a macro command: double-click it to open the Macro Shortcut window.
- Delete a macro command: select it and then either press the **CTRL + DELETE** keys, or select **Delete Macro** from the context menu.
- Remove a macro command shortcut: select the macro and then either press the **DELETE** key, or select **Remove Shortcut** from the context menu.

Macro Shortcut

The New/Edit Macro Shortcut window is opened by selecting Add Macro from the Keyboard Configuration Window context menu, or double-clicking an existing macro.

The Macro Shortcut window contains the following fields and buttons:

- **Enter Key:** Sets the keyboard shortcut for the macro.
- **Macro Name:** Displays the macro category (for example macros:, gui:, viz:) and the name of the macro. If a folder was selected when the Macro was opened, the category name is automatically displayed in the Name text box. The category name can be set or changed in the Name field by adding the category and macro name separated by a colon (<macro category>:<macro name>).
  - Note: If a category is not added to the name it will be placed in the macros category. It is also possible to add new categories by defining them in the macro (for example foo:bar).
- **Global:** Makes the macro work even if no playlist is open.
- **Description:** Description of the macro.
- **Macro Commands:** The names of the selected Predefined Functions are displayed.
- **VB Script:** A text field where script code can be entered. This provides the possibility to further enhance and adapt the commands.

> ⊗ **IMPORTANT!** The function stub named *RunScript* must be used as the main script or entry point for other scripts. It can therefore not be renamed or removed.

- **Show Macros:** Opens the Predefined Functions window.
- **Save As New:** Saves the macro as a new instance.
- **OK:** Saves the macro, either as a new or modified macro.
- **Cancel:** Closes the window, without saving any changes.

Predefined Functions
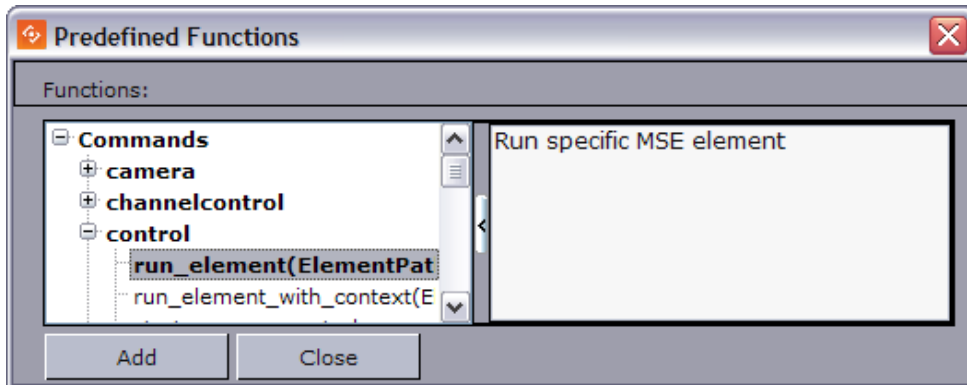
Tip: User information for each language is stored under `%ProgramFiles(x86)%\Vizrt\Viz Pilot x.y\scriptdocs`

8.3   Director Preferences

Selecting Preferences on the Options menu opens the Preferences window. The Preferences dialog is used to set parameters for Director and the Viz Pilot database.

This section contains the following topics:

- Startup Preferences
- General Preferences
- Viz Connection Preferences
- Playlist - General Preferences
- Playlist - Colors Preferences
- Playlist - Filters Preferences
- Remote Preview Preferences
- Viz World Preferences
- Spellcheck Preferences
- Resource Plugins Preferences

8.3.1   Startup Preferences
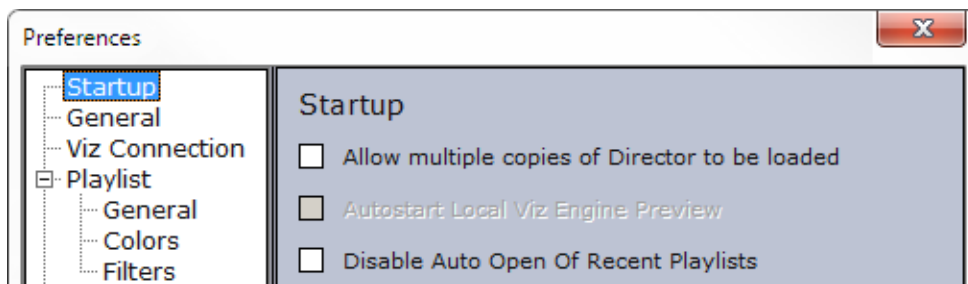
- **Allow multiple copies of Director to be loaded:** If enabled, multiple copies of Director can be loaded simultaneously.
- **Autostart Local Viz Engine Preview:** If enabled, a local instance of Viz Engine will automatically start when Director is launched.
- **Disable Auto Open Of Recent Playlists:** When checked, any opened playlists will not be opened when restarting Director.

8.3.2 General Preferences
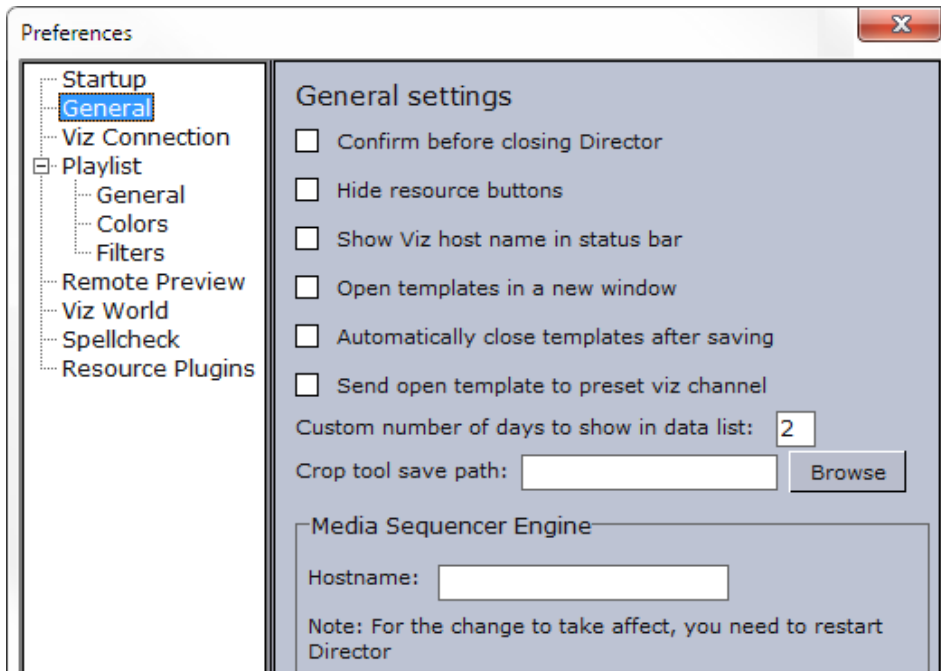
- **Confirm before closing Director:** If enabled, the user must confirm that Director should close.
- **Hide resource buttons:** If enabled, the buttons on the Buttons panel will be hidden. The different views can still be selected through the Drop-Down list.
- **Show Viz host name in status bar:** Shows the Viz Engine hostname on the status bar.
- **Open templates in a new window:** If enabled, the template will be opened in a separate window.
- **Automatically close templates after saving:** If enabled, templates will be closed automatically when saved.
- **Send open template to preset viz channel:** Sends the open template to the preset Viz Engine channel.
- **Custom number of days to show in data list:** Shows data items that have been used the last X number of days.
- **Crop tool save path:** Sets the path for the image saved when using the Crop tool in Viz Pilot News. See Image Component.
- **Media Sequencer:** Sets the hostname of the Media Sequencer machine.

8.3.3 Viz Connection Preferences
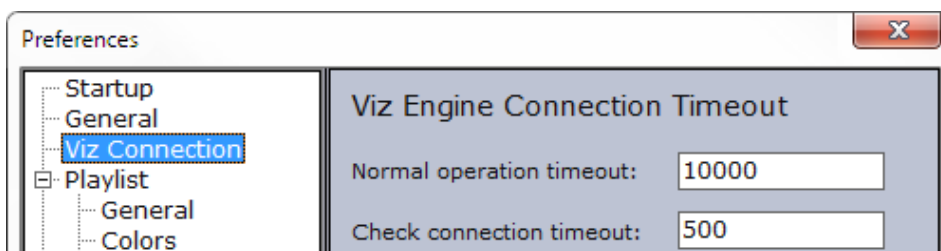
- **Normal operation timeout:** An error report will be displayed if Viz does not send a response within X ms when a user is browsing for data from Viz. By default, the Normal Operation Timeout value is set to 10.000 ms (10 sec).
- **Check connection timeout:** Right-clicking the Viz Connection icon on the Status bar makes the Check Connection operation available. If Viz Engine does not respond within X ms after checking the connection, the Viz Engine connection icon turns red. By default the Check Connection Timeout value is set to 500 ms.

8.3.4   Playlist - General Preferences
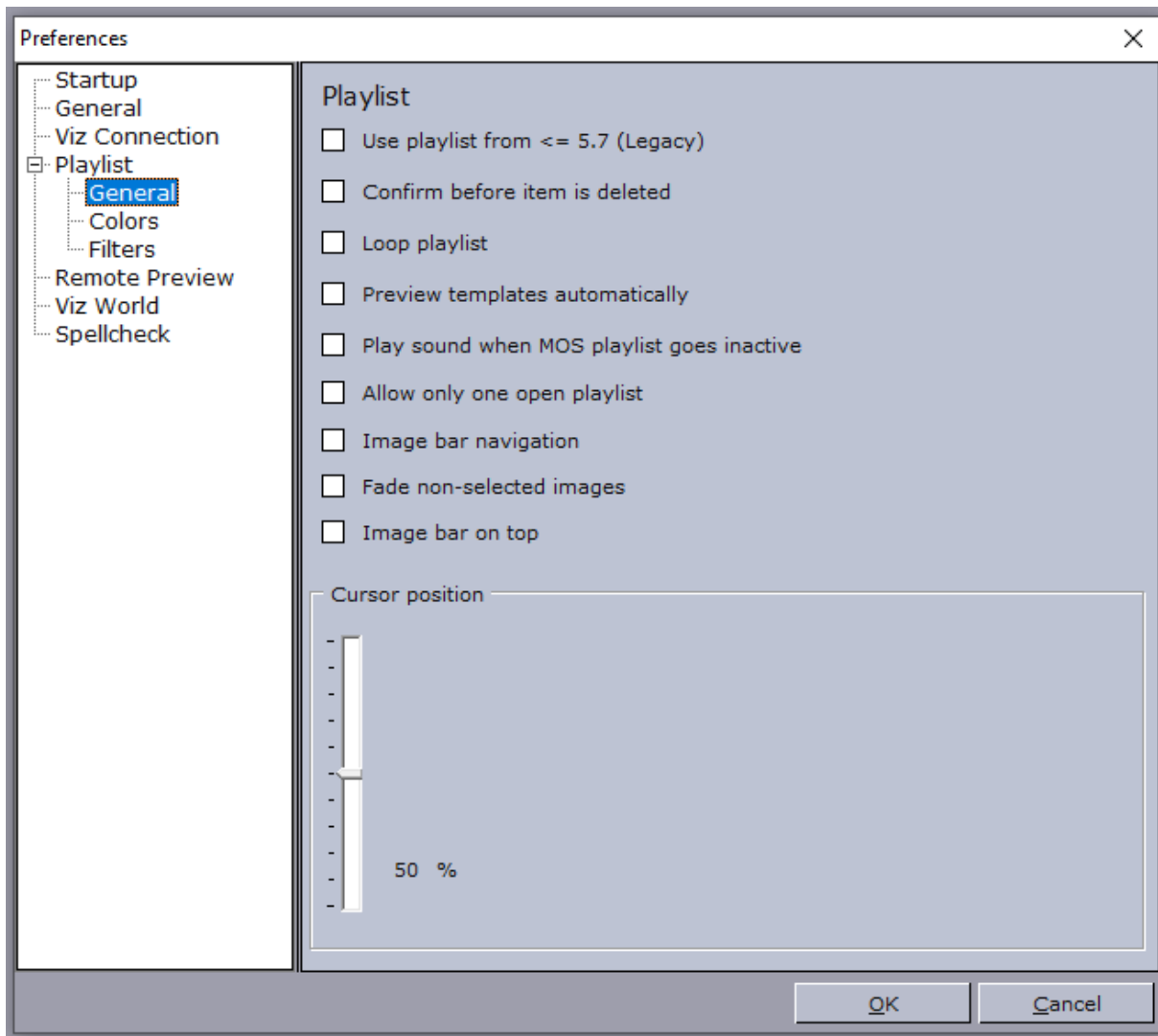
- **Use playlist from <= 5.7 (legacy):** If enabled, the old (legacy) playlist will be used. Otherwise, the new playlist will be used.
- **Confirm before item is deleted:** If enabled, the user must confirm that the delete operation should be completed.
- **Loop playlist:** Loops all the elements in the playlist.
- **Preview templates automatically:** Opens the selected element in the playlist automatically.

- **Play sound when MOS playlist goes inactive:** Plays a sound whenever the newsroom (MOS) playlist becomes inactive.
- **Allow only one open playlist:** Ensures that only one playlist can be open at a time. When opening a playlist, other open playlists will be closed.
- **Image bar navigation:** Selects the corresponding element in the playlist when a thumbnail on the **Image** bar has been selected.
- **Fade non-selected images:** Fades all images, except for the selected thumbnail.
- **Image bar on top:** Shows the **Image** bar at the top of the playlist.
- **Cursor position:** When the playlist's cursor is active (see Legacy Playlist Context Menu), this setting determines where in the window the cursor stays in the situations described above. The position can be adjusted by moving the slider control.

8.3.5   Playlist - Colors Preferences

> ⚠ **Note:** These settings are only relevant for the **legacy playlist**. The playlist available with Viz Pilot 6.0 and later does not use background colors, but other, non-configurable visual clues for a subset of the below categories. To select which playlist you will use, go to Playlist - General Preferences.
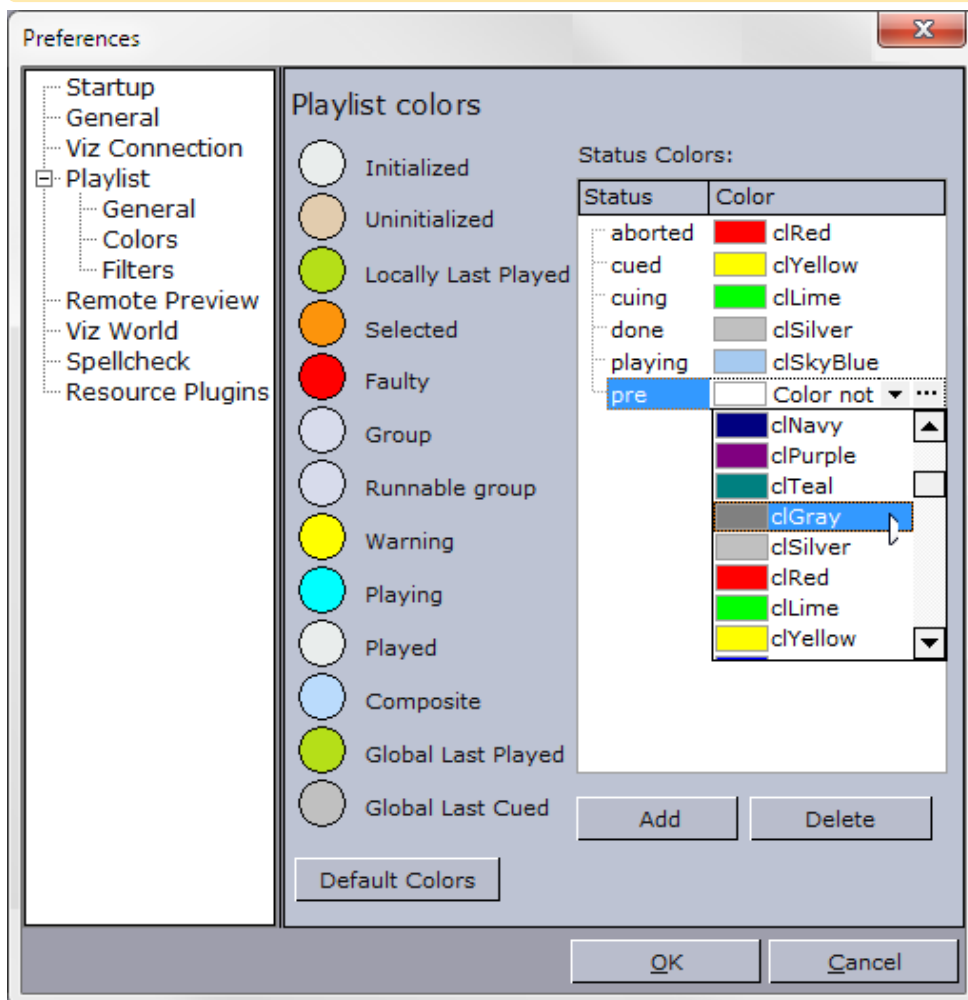
The Colors window displays status colors of the different elements in the playlist. The element colors can be customized. To change the color of an element, click the colored circle adjacent to an element label. In the Color window that opens, pick a color, then click **OK**.

Some important Playlist Colors that can be configured are:

- **Runnable group:** Refers to timeline editor groups (runnable groups). Makes it possible to color video timeline groups in the playlist differently than other groups.
- **Global Last Played/Cued:** Refers to data elements played or cued by the Media Sequencer based on commands issued by other control clients or through a GPI.
- **Default Colors:** Loads the default color settings.

The **Status Colors** list refers to the Status column in the playlist. To add a color to a Media Sequencer status, click the Add button. Select the Status column of the new entry that appears in the Status Colors list, and then type the appropriate status name. Then select the Color column of the entry. In the drop-down list that appears, select a color.

- **Add:** Adds a new color to a Media Sequencer status.
- **Delete:** Deletes a color.
  Alternatively, click the ellipsis (**...**) button to select a color using the Color Picker.

> ⚠ **Note:** You must close and reopen playlists for the color change to take effect.

When creating a category in Template Wizard, the category you create can be used to set which category a template belongs to, and on which channel it will subsequently be played out. Note that the channel name configured in Template Wizard must correspond with the profile configured for Director. See Categories and Channels.

A category can be used to filter the different template variants. For example a Lower third (straps) filter, or a fullscreen filter. In some cases the graphics are played out on different renderers. Thus the channel configuration adds a usable feature to differentiate the output.

To filter by category in the Playlist Window, select **Subcategory** in the Playlist Filter and Filter Editor.

8.3.6   Playlist - Filters Preferences

> ⚠ **Note:** These settings are only relevant for the legacy playlist. Filters for the new playlist can be configured by using the filter button on the toolbar of the playlist. To select which playlist you will use, go to Playlist - General Preferences.
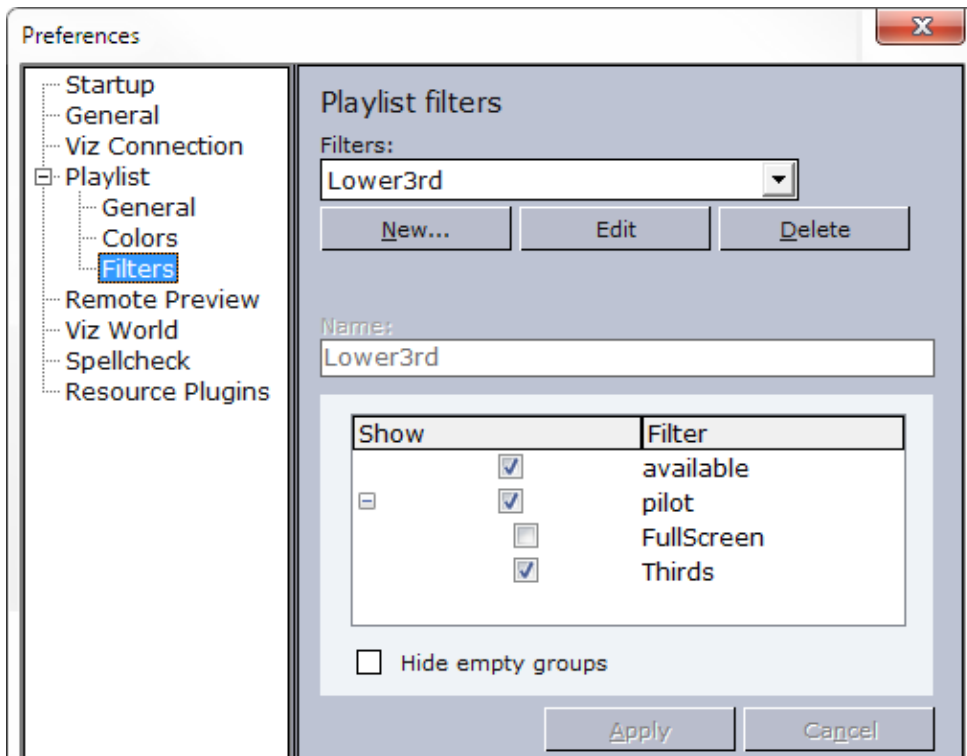
- **Filters:** Displays the available operator defined filters.
- **New:** Makes the categories and Name field available to create a new filter.
- **Edit:** Makes the settings belonging to the selected filter editable.
- **Delete:** Deletes the selected filter.
- **Name:** Displays the name of the filter.
- **Hide empty groups:** Hides empty groups.
- **Apply:** Applies the new or modified filter settings.
- **Cancel:** Cancels the recent filter settings.

The image below shows a playlist with three data elements:
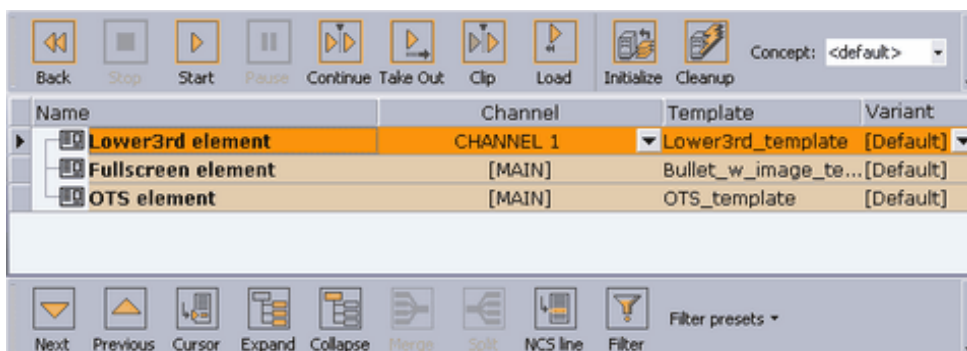
The image below shows the same playlist with the Lower3rd filter (see Legacy Playlist Toolbar) enabled:
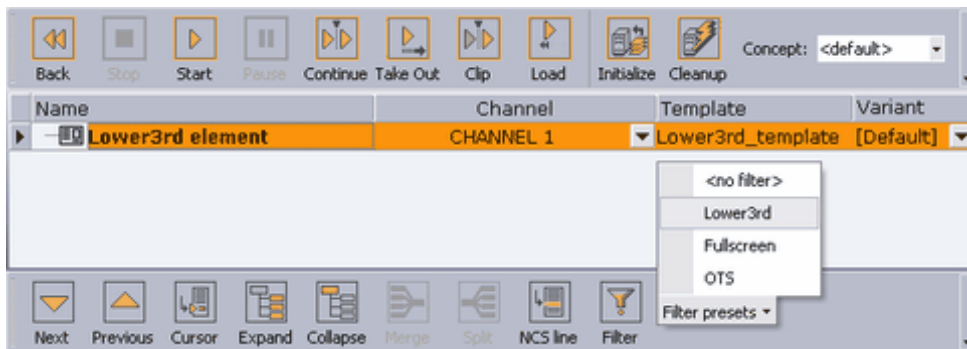
Creating a New Playlist Filter

> ⚠️ **Note:** Create categories and channels in Template Wizard.

1. Select the Director Preferences option on the **Options** menu.
2. Select **Filters** under the **Playlist** category.
3. Click **New...** to add a new filter.
4. Select the category, or categories, which the filter applies to.
5. Click **Apply**, and confirm the dialog to save the new filter.

> ⚠️ **Note:** All category and channel configurations are saved to the Viz Pilot database. Filter configurations are saved to the Media Sequencer.

> ❌ **Warning:** If the Media Sequencer is reset, all filter settings are removed.

8.3.7   Remote Preview Preferences
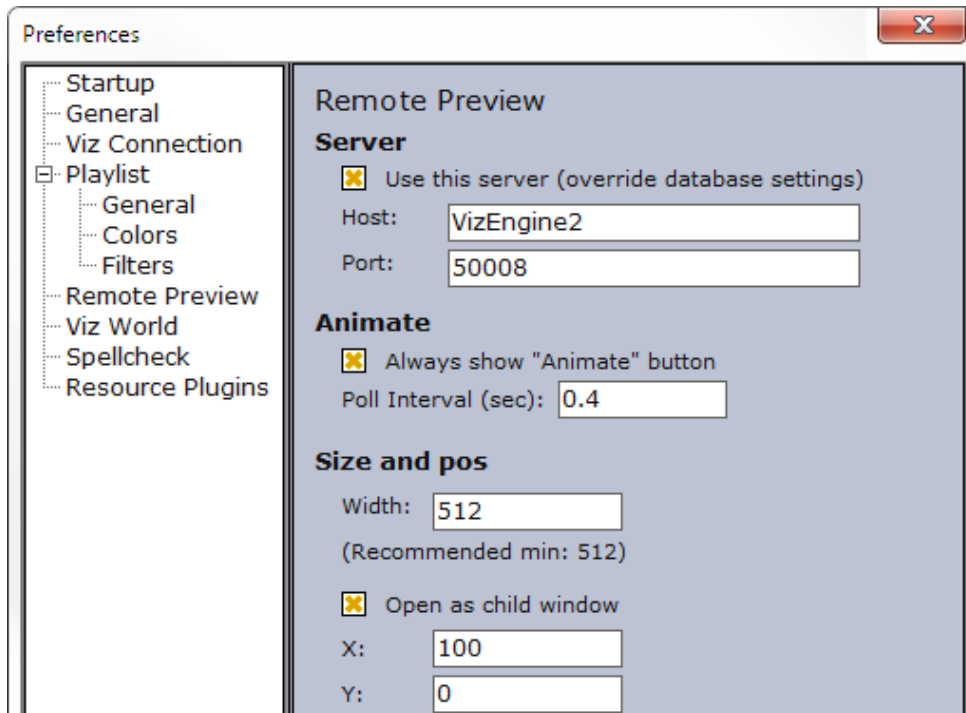
Remote preview is a feature that is used to view snapshots of scenes rendered on a remote Viz Engine, and should not be mistaken for the preview channel.

> ⚠ **Note:** If the Viz Engine Preview plugin is enabled, local preview starts instead of remote preview (snapshots). The plugin can be disabled in the Changing Settings.

- **Use this server:** Enables the user to set host and port parameters locally in order to override the remote preview configurations on the database.
- **Host:** Defines the external Viz Engine host name.
- **Port:** Sets the external port number. Recommended port is `50008` .
- **Always show "Animate" button:** Enables the **Animate** button in the **Remote Preview** window.
- **Poll Interval (sec):** Sets how often Viz Engine is polled for a new snapshots of the previewed scene.
- **Width:** Sets the width of the preview window.

> ⚠ **Note:** For optimal preview, use the Viz Engine Output Format as a preference.

- **Open as child window:** Opens the window within the Work area.
- **X:** Sets the X coordinate. *0* is the value in the upper left corner.
- **Y:** Sets the Y coordinate. *0* is the value in the upper left corner.

For more information, see Configuring Remote Preview for the Newsroom.

8.3.8   Viz World Preferences
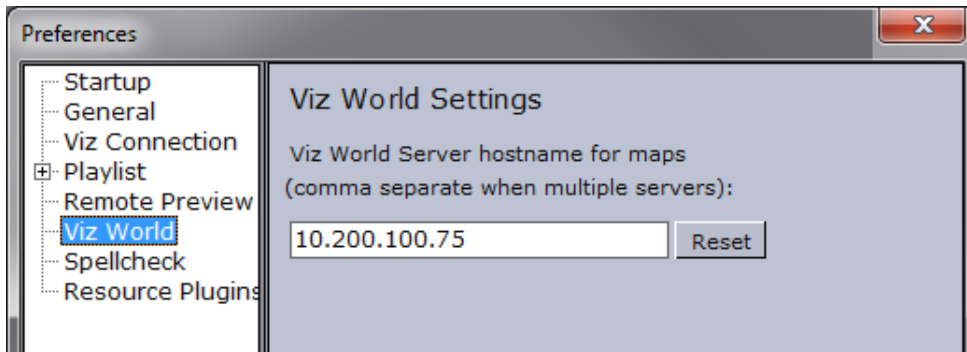
- **Viz World Server:** Sets the hostname or IP address of the Viz World Server.

8.3.9   Spellcheck Preferences
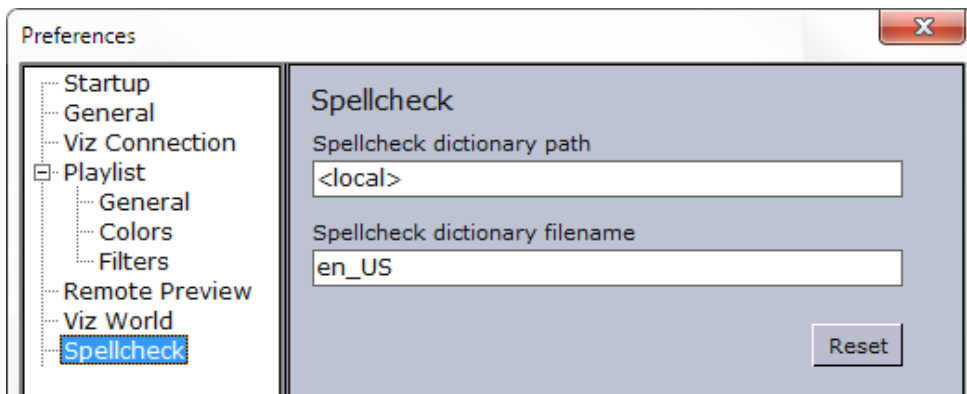
- **Spellcheck dictionary path:** Sets the path to the dictionary files for the spell checker. If empty, the default dictionary location is used: `%ProgramFiles(x86)%\vizrt\Common\dicts`
- **Spellcheck dictionary filename:** Sets the filename of the dictionary files. For example *en_US* refers to the affix file *en_US.aff* and dictionary file *en_US.dic*.
  For more on setting these parameters, see **spellcheck_dict_path** and **spellcheck_dict_filename** in the Database Parameters.

> ⚠️ **Note:** Spell checking can only be enabled for TTWUniMemo (Unicode Memo Component) and TTWUniEdit (Unicode Edit Component) components.

8.3.10   Resource Plugins Preferences
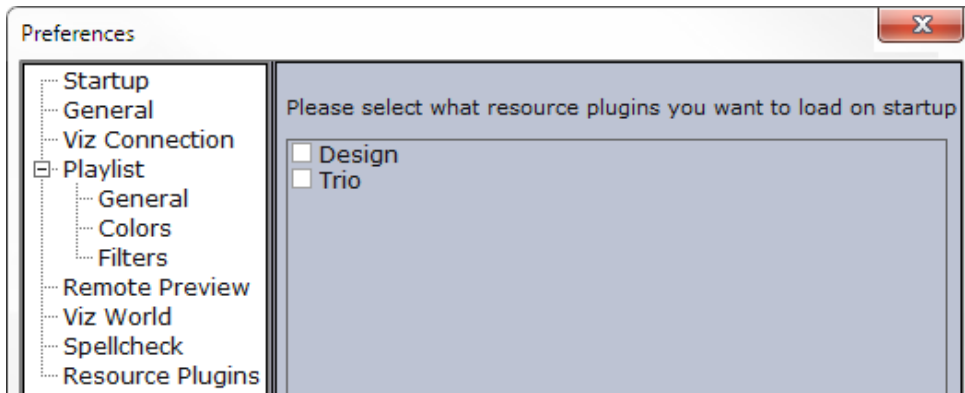
- **Trio:** Enables the Trio resource panel.

8.4 Director Tools

This section contains the following topics:

- View Log
- Show Command Console
- Post Render Device
- Timecode Monitor

8.4.1 View Log
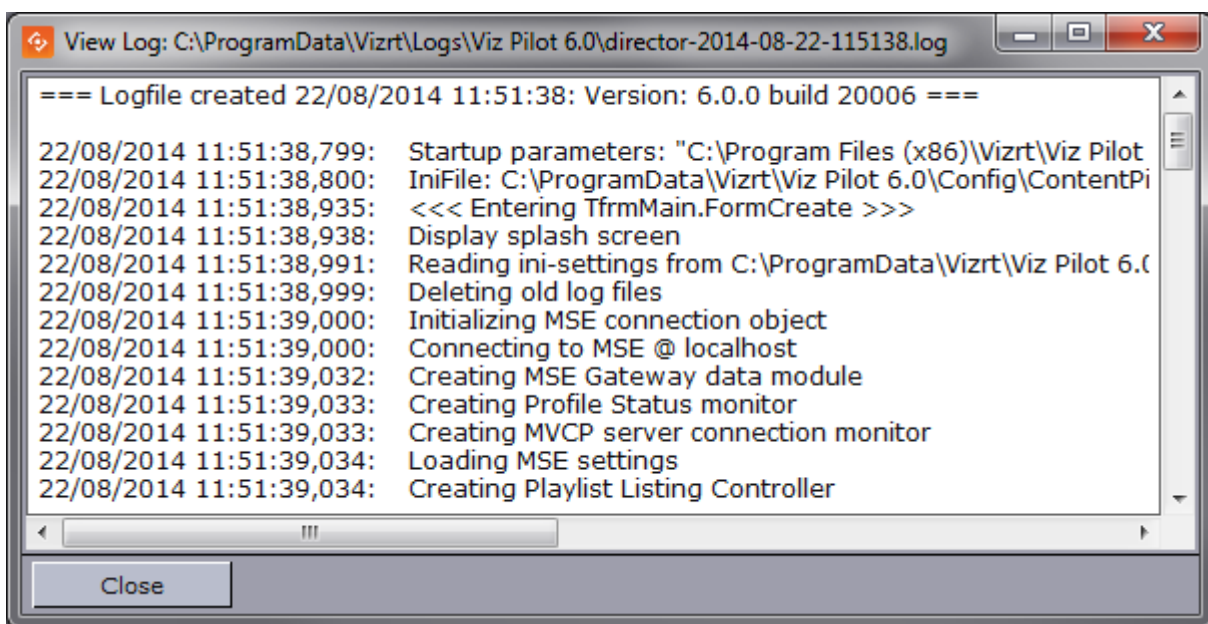
Open from the main menu: **File > View Log**.

The View Log window displays the log messages that have been generated by the system. Individual log messages are also displayed as Status Information in the Errors Window.

> ✅ **Tip:** Check the View Log window frequently when troubleshooting the system.

The full log files can be accessed. Default log file locations are as follows:

- `%PROGRAMDATA%\Vizrt\Logs\Viz Pilot x.y`
  Eureka Log files (.elf crash log files) for Director, Template Wizard and Viz Pilot News can also be found in this directory.

8.4.2   Show Command Console
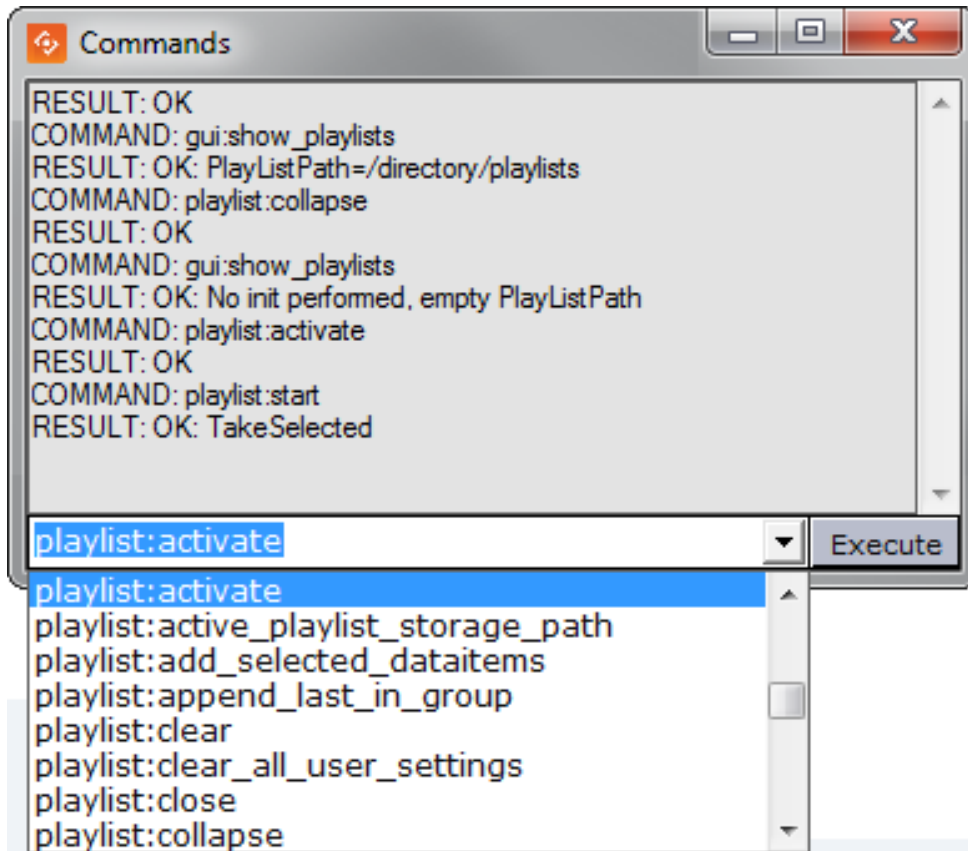
The Commands window can be used to see the macro commands that are executed when different user operations are performed. The window is also helpful in learning the different macro commands.

You can show commands by selecting **Show Command Console** from the **Tools** menu.

To run a command you must enter its name (for example `gui:new_playlist`) in the combo box, and then click the **Execute** button to run the command.

Looking Up Commands Using Apropos

The Commands window has a lookup function named **apropos** that is helpful when searching for information about a command.

1. Open the **Commands** window.
2. Enter for example `apropos run_element` and press **Enter**. The following result should appear:

```
COMMAND: main:apropos run_element
RESULT: control:run_element(restString ElementPath) : Run specific MSE element
control:run_element_with_context(string ElementPath, string ContextName, string
ContextValue) : Run specific MSE element where ContextName is a comma separated
list of Names and ContextValues their values
```

8.4.3   Post Render Device

Post rendering is used to create images or video files of graphical scenes. The files can be used for playout on Viz Engine. Selecting a video plugin will create one file; however, selecting an image plugin will render an image according to the configured frame rate. For example: Rendering a scene for five seconds will result in 125 images if the frame rate is 25 frames per second (fps).

Rendered data elements can be fullscreen graphics or graphics with Alpha values such as *lower thirds* and *over the shoulder* graphics.

> ⚠ **Note:** Only scene-based data elements can be post rendered.

This section contains the following topics:

- Post Rendering Window
- Post Rendering a Scene
- Setting Up a Render Device

Post Rendering Window
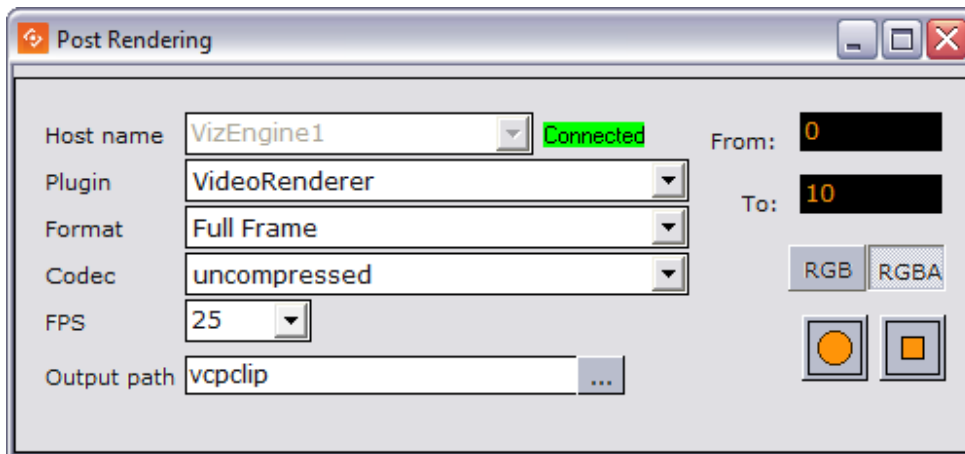
The post render devices are set up automatically and do not require any configuration. Any changes to the device settings will be maintained between recordings.

- **Host name:** Displays the render engine that is configured in the profile configuration. To change the render engine, open the Profile Configuration.
- **Plugin:** Sets the renderer device. Options are defined by the installed codecs on the render engine.
- **Format:** Sets the format of the clip. Available options are **Full Frame**, **Fields Top**, **Fields Bottom**, **Full Frame/Interlaced Top** and **Full Frame/Interlaced Bottom**.
- **Codec:** Sets the codec to be used.
- **FPS:** Sets the frames per second. Available options are 25, 30, 50, 60, 29.97 and 59.94.
- **Output path:** Sets the filename. A full or relative path can be added. If no path is given, the file is stored in the Program Files folder, or, for Windows 7 and later, the VirtualStore folder on the rendering machine (%LOCALAPPDATA%\VirtualStore\Program Files\Vizrt).
- **From:** Sets the start time in seconds.
- **To:** Sets the end time in seconds.

- **RGB:** Sets the pixel format to RGB.
- **RGBA:** Sets the pixel format to RGBA which includes the alpha channel (blending/transparency).
- **Record button (circle):** Starts the rendering process of the selected data element. Clicking the **Record** button will run the recording process according to the duration interval set using the **From** and **To** parameters. Controlling the data element is possible during the recording.
- **Stop button (square):** Stops the rendering process before the configured stop time.

Post Rendering a Scene

1. Open a data element.
2. Open the **Post Rendering** window.
3. Set the output parameters.
4. Click the **Record** button.

Setting Up a Render Device

Rendering is automatically configured - no setup is required.

> ⚠️ **Note:** For more information on how to use the Device Manager in Viz Artist, see the *Viz Artist 3.x User Guide*.

8.4.4 Timecode Monitor
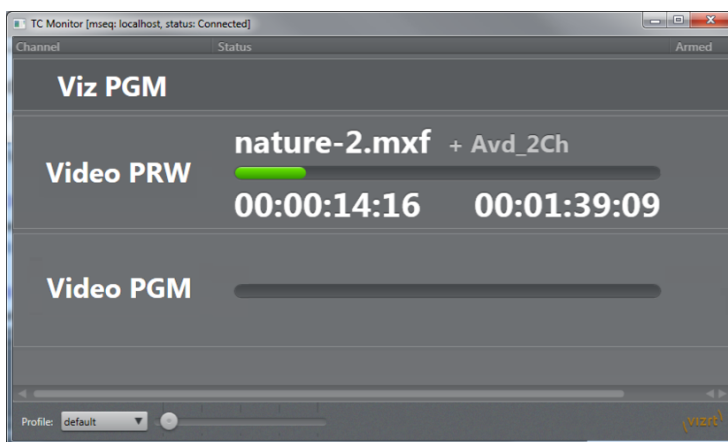
Selecting **Timecode Monitor** on the **Tools** menu opens the **TC Monitor** window. Since it monitors the timecode and the video server channels, the Timecode Monitor will only work if the system is configured to be integrated with a video server or Viz Engine. Director uses the Multiport Video Control Protocol (MVCP) to communicate with a translator (Xlator) which in turn communicates with the video server.

> ⚠️ **Note:** For more information on supported video server setups, please contact Vizrt.

The Timecode Monitor displays three columns: Channel, Status and Armed.

- **Channel:** Name of the Channel. Video channels appear as light grey rows, with progress information in the Status column. Non-video channels are shown as dark grey rows without status.

- **Status:** The name of the current item, followed by any cued item, is displayed above the progress bar. Current Location time and Remaining time are shown below the progress bar. The progress bar changes color to indicate status:
    - *Green:* Video playing.
    - *Red:* Video playing, less than 10 seconds remaining.
    - *Blue:* Video finished.
- **Armed:** Displays the armed state of an element on a channel. This feature is not used in Director.
    The rows in the list can be shown or hidden using the context menu on each item:

- **Hide:** Hides the selected channel.
- **Show All:** Shows all channels.
    At the bottom of the window is the zoom function to adjust the display size, and the Profile selector which displays the current profile name, and allows selection of other profiles.

The Timecode Monitor opens in full screen by default. Press the **ESC** key to exit full screen mode.

> ⊗ **IMPORTANT!** In order to monitor video clips that are embedded in graphics, the Viz channel must also be defined as a video device channel.

See Also

- Log Files

8.5 Viz Engine Preview In Director

Local preview uses a local Viz Engine, while remote preview uses a remote Viz Engine.

- Local Preview in Director
- Remote Preview in Director

8.5.1 Local Preview in Director
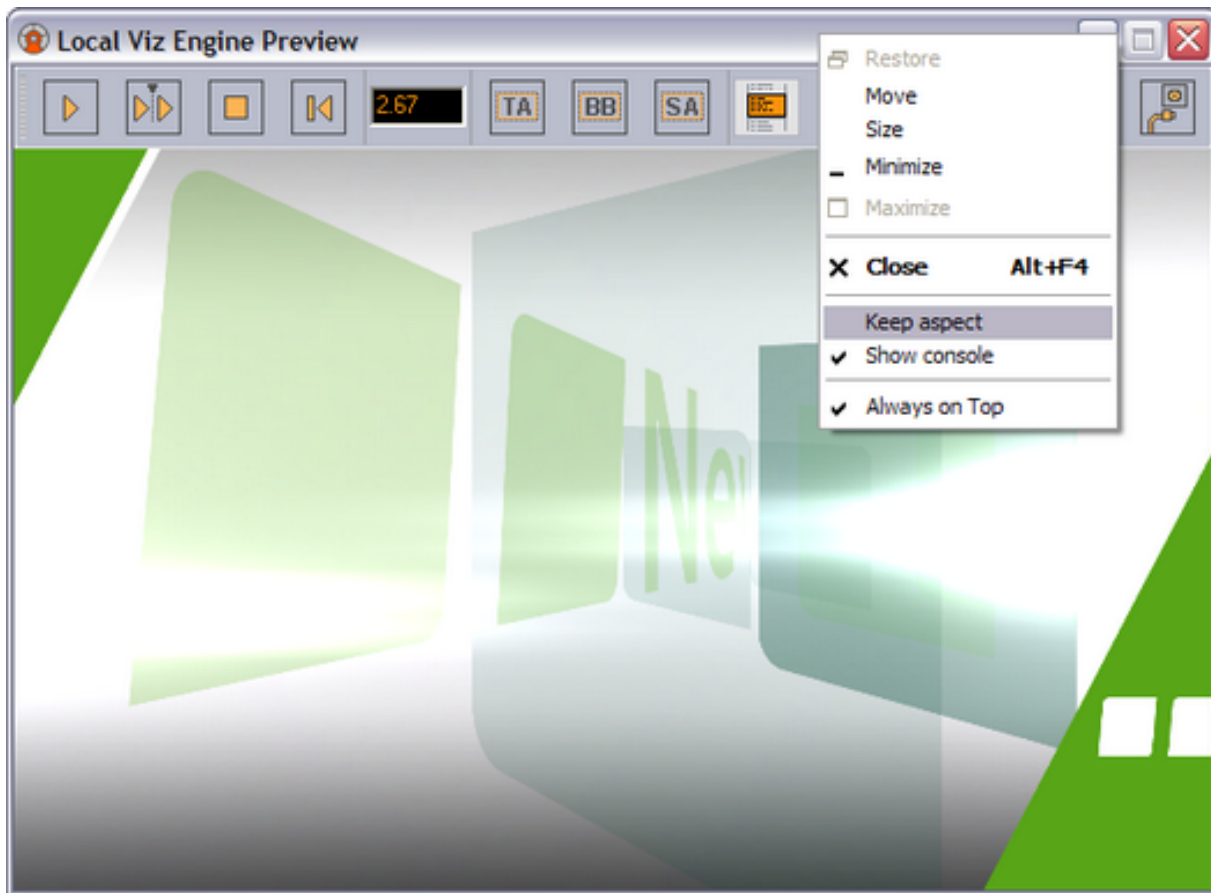
Local Preview means that a Viz Engine is installed locally on the Director machine for preview purposes. It also requires a Viz Engine Preview Plugin that is not enabled by default. The plugin can be enabled/disabled using the Changing Settings. See Configuring Local Preview.

The interface is the same as that used in Viz Pilot News, see Local Preview in the Newsroom.

To See a Local Preview

1. Set a local Viz Engine (localhost) as the preview renderer in the Profile Configuration.
2. From the main menu select **Viz Engine** > **Local Preview**, or press **CTRL + P.**
3. Open a playlist.

4. Right-click the playlist and check the **Auto preview** option (which allows a preview to be sent as soon as an item in the list is selected).
5. **Click** a data element and it will be sent to the preview window.

8.5.2   Remote Preview in Director

Remote Preview means that a Viz Engine is installed remotely on another machine for preview purposes.

This option lets you see a remote snapshot preview (still or animated) of a template or data element. Animated means that a series of frames are fetched from the preview renderer on-the-fly; hence, a snapshot preview is not a real-time rendering. However, if you have a reference monitor connected to your Viz Engine's DVI/VGA output, you may use the preview window's control buttons to play or scrub the graphics in real-time.

> ⚠ **Note:** To configure Remote Preview, see Configuring Remote Preview in Director.

Properties and Parameters

- **Start:** If the graphic has two or more tags on its timeline, this asks Viz Engine for a snapshot of the first frame or the first tag on the main (default) director. If the scene, or foreground scene in transition logic terms, has a **pilot1** tag it by default jumps to that tag on the director and shows that frame.
- **Continue:** If the graphic has two or more tags on its timeline, this asks Viz Engine for a snapshot of the remaining tags on the default director.
- **Key:** Shows the key signal of the graphics.

- **Time field:** Shows the current timeframe for the scene. Allows the user to scrub the scene manually.
- **Tag field:** Shows the scene's director tags on the Default director. These tags are used for previewing the scene, and are not actual stop points. Proper stop points enable the use of Continue to trigger the animation and should not be used for creating tags for preview purposes.
- **Animate:** Asks Viz Engine for a series of snapshots according to the Remote Preview Preferences configuration under Director Preferences; this in turn is played out in sequence in the preview window in order to animate the scene.
- **Title Area:** Shows the boundaries of the defined title area (green rectangle).
- **Safe Area:** Shows the boundaries of the defined safe area (purple rectangle).
- **Copy:** Copies the current snapshot onto the Windows clipboard.
- **Save:** Saves the snapshot as a PNG file.
- **Error:** Shows the error log for the current session.
- **Connection:** Shows the current Viz Engine connection status.

To See a Remote Snapshot Preview

1. Right-click in the playlist and from the context menu select **Auto Preview**.
2. From the main menu select **Viz Engine** > **Remote Preview**, or press **CTRL + P**.
3. Open an item in the playlist. This will send a preview to Viz asking for the first frame of the scene or the frame of the first preview tag.

> ⚠ **Note:** When using the Remote Engine Preview, users may need to log in with valid credentials more than once.

See Also

- Viz Engine Preview Configuration
- Viz Engine Preview in the Newsroom

8.6   Status Bars
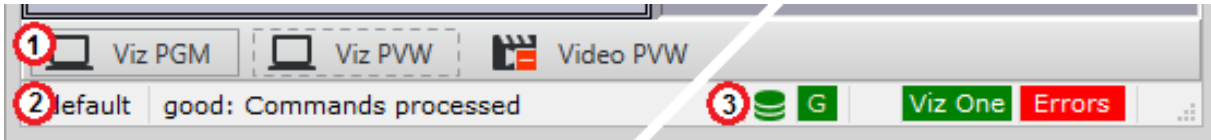
The Status bar at the bottom of the window consists of:

1. Profile and Channel Information.
2. Status Information.
3. Status Indicators.

8.6.1   Profile and Channel Information

Profile Context Menu
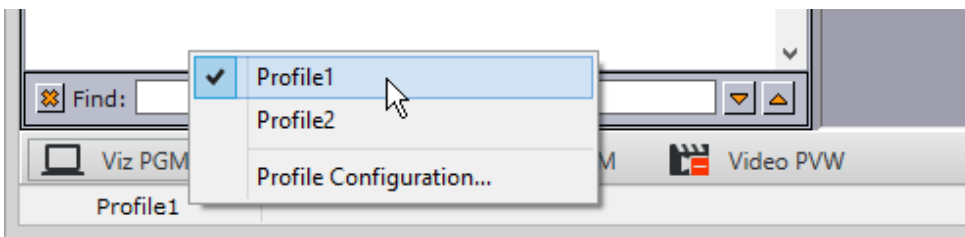
Right-clicking the profile name will open the Profile Context Menu:

- **Profile(s):** Shows the available profile name(s), and indicates the current profile.
- **Profile Configuration:** Opens the Profile Configuration window.

Channel Context Menu
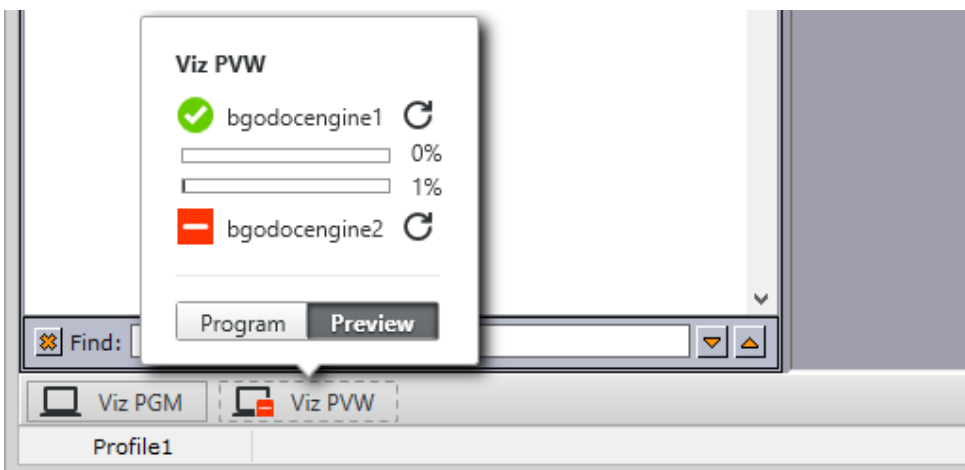
Right-clicking Viz Channel will display a context menu for setting the channel as a program or preview channel. In addition, Viz Engines that are configured for that channel and their current status it will be displayed.

Clicking the refresh icon next to the Viz Engine name refreshes the status indicator and current memory usage bars.

Clicking the Channel indicator when it is disconnected (status: red) makes the system try to reconnect (see the Profile Configuration window).

8.6.2 Status Information

Status information is positioned to the lower left.

Profile1	Active profile - right click to change
Profile1	MSE Database connection : Connected
Profile1	error: connection failed, please check settings.
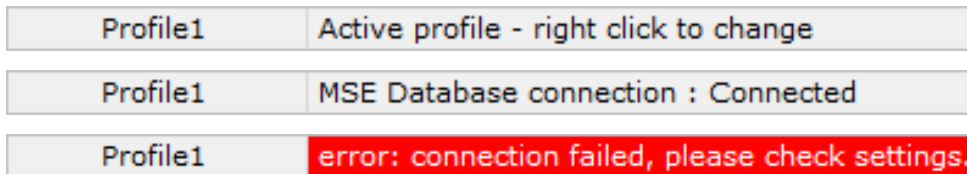
Status information provides feedback on different objects in the Director user interface and from connected systems, both:

- **General information:** General information is displayed when a pointer is positioned over objects in the user interface that contain information relevant to the user. When the Hints area displays general information, it remains its default color (grey).
- **Error information:** Error information is displayed if errors occur. When the text field displays error information, the text field background color turns red. Error information is also displayed in the Errors Window.

8.6.3 Status Indicators
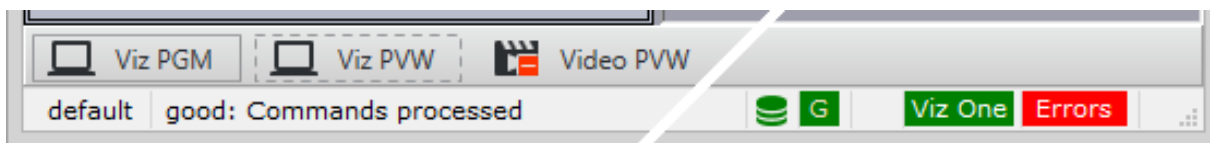
Status Indicator Items

Item	Description
	Media Sequencer's database connection: Indicates status of the Media Sequencer's connection to a Viz Pilot database. Hover over the icon to display more information about the connection. *Green*: The Media Sequencer is connected to a Viz Pilot database (see Media Sequencer Configuration). However, note that the indicator is green even if the Media Sequencer is connected to a different database than the one Director is connected to (for example, Director is connected to Graphic Hub, while Sequencer is connected to Oracle). *Red/Orange*: Indicates a problem with the Media Sequencer's database connection. Check the configuration by double clicking on the icon or check the Media Sequencer logs. *Gray*: The Media Sequencer is not configured to connect to a Viz Pilot database, or information about the status of the connection is currently unavailable. Check the Media Sequencer's configuration by double clicking the icon.

Item	Description
G	**Viz Gateway:** Green if the Media Sequencer is connected to Viz Gateway. Red if there is no connection. Clicking the indicator when it is disconnected (status: red) makes the system try to reconnect (see Media Sequencer Configuration).
M	**Matrix:** Green if Director is connected to Viz IO. Red if there is no connection. Clicking the indicator when it is disconnected (status: red) makes the system try to reconnect.
Viz One	**Viz One:** Green if the Media Sequencer is successfully connected to Viz One. Red if the configuration is not working properly (for example if a Viz One search fails). In serious situations where the Viz One is unavailable, an additional pop-up error message will appear. In that case, videos that have already been transferred to Viz will still work, but any new videos will not be transferred. Double-clicking the indicator opens the Media Sequencer Configuration window.
Errors	**Errors:** Turns red when an error occurs. Clicking the button opens the Errors Window. Errors that occur are saved in the Errors log, and displayed in the Errors window.
	Viz Channel: Displays the status of the Viz channel's Viz Engine(s). See Profile and Channel Information.
	Video Channel (clapperboard): Shows whether the Media Sequencer is successfully connected to a Viz Video Engine, Viz Engine or other video server for video playout. See Profile and Channel Information.

Errors Window
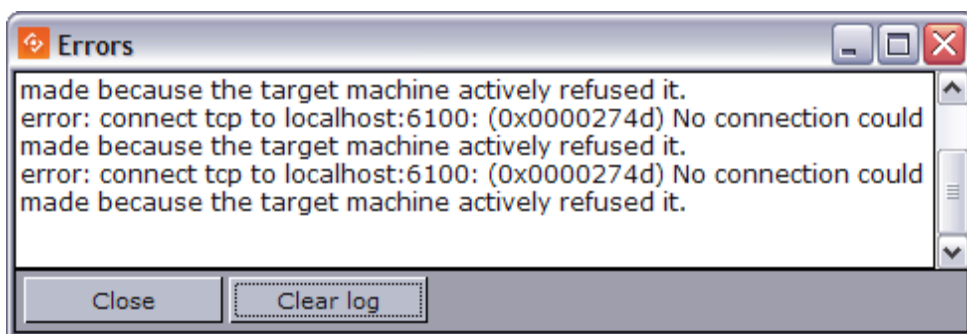
- **Clear Log:** Removes all the error messages from the log window, but not from the log file. The Errors window also closes, and the Errors button is hidden until there is new error log information to display.

8.7   Resource Panel
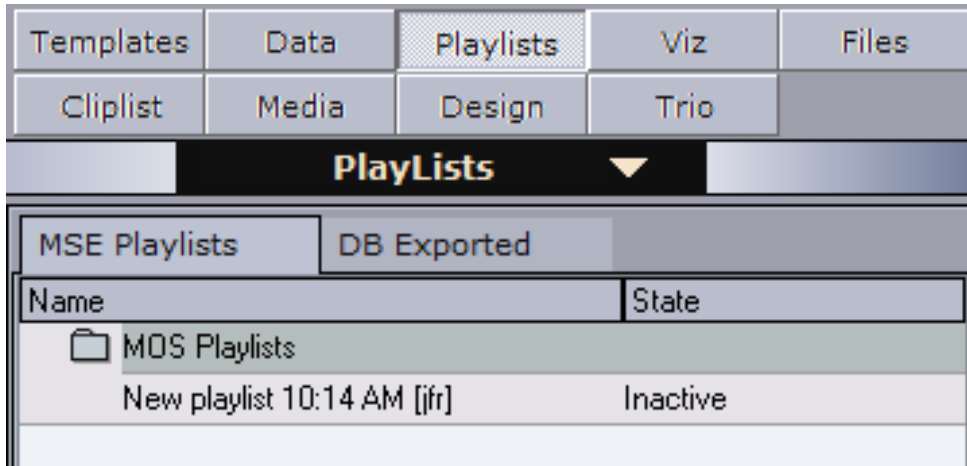
Director can be configured to access up to nine different resource pools. The resource panel is positioned to the left in the main program window. The buttons and the drop-down menu can be used to open resource pools.

- Templates and Data provide access to all graphics templates and data elements stored in the Viz Pilot database. Data elements are templates that are filled out by users of the system (such as journalists or operators).
- Playlists provides access to all playlists stored on the connected Media Sequencer. They are either manually created, or MOS monitored by Director when created from a Newsroom Computer System (NCS).
- Viz provides access to elements on the Graphic Hub (via a Viz Engine).
- Files provides access to the local computer's file system. This must be used with caution as selected files need to be accessible from all devices that use them (including Viz Engines).
- Cliplist connects to video servers over the MVCP protocol.
- Media provides access to the media repository of Object Store and Viz One. The Media tab enables the operator to search for and add image and video items directly into the playlist or to the templates/data elements through drag and drop.
- Trio provides access to Viz Trio's show folders and shows that are stored on the same Media Sequencer that Director is using. Accessing the shows enables the user to drag and drop Viz Trio pages into a Director playlist.

The **Design** tab is not in use.

> ⚠ **Note:** Audio search cannot be used in a Director system.

8.7.1   Resource Search

 Typing the first letters of the name of a resource in the search area marks the resource that correspond to the letters in the list.

The resource search applies to Templates, Data, Playlists and Trio.

> ⚠️ **Note:** The resource search is always enabled for the Viz Trio elements tab.

8.7.2 Show the Search Area

1. Right-click a resource list, and select **Find**, or
2. Press the keyboard shortcut keys **CTRL + F.**
3. Enter the search criteria and navigate up and down with the arrow buttons to find the next or previous resource matching the search criteria.

8.8 Templates

Templates are built using Template Wizard. Templates are used to add content and edit the look and feel of graphics.

This section contains the following topics:

- Template Panel
- Working with Templates
- Concepts and Variants
- Template Control Buttons
- Template Save Dialog Box
- Template Spell Checker
- Timeline Editor Preview

8.8.1 Template Panel
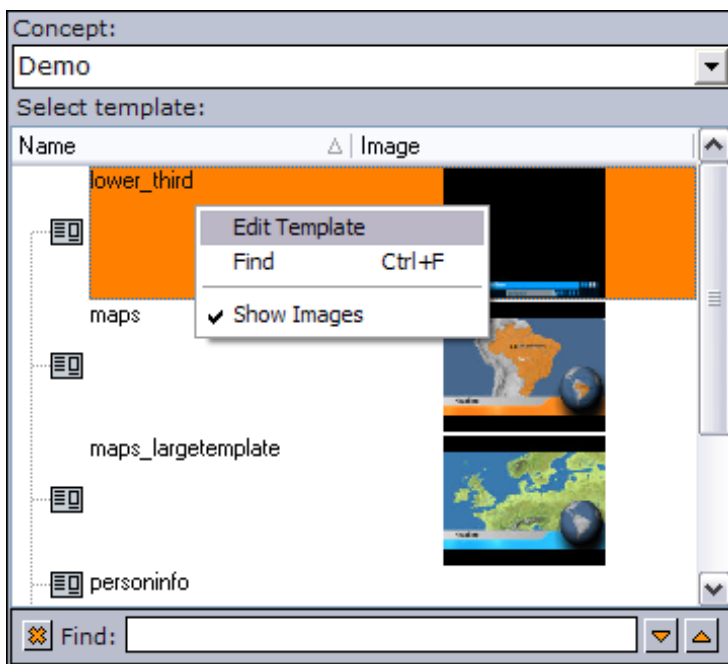
Data stored on the Viz Pilot database with references to a template and scene is called a Data element when its displayed in Director. Data elements added to Playlists are stored on the Media Sequencer for playout, and are consequently stored with references to the templates and scenes.

A template is used to populate one or many data elements in order to have elements with differing content (for example text, images, video), and differing control parameters than other Data elements.

Additionally, a template, and consequently a data element, may belong to one or more concepts (for example Sports and News scenes) and belong to one or more variants of a scene (for example Sports lower and top thirds).

> ⚠ **Note:** All (<All>) templates are listed the first time Director is started. The next time Director is started it will remember which concept was selected and start with that concept template listing.

Context Menu - Templates

Context menu for the columns:

- **Auto Width:** When enabled (selected) Auto Width expands the columns and distributes them evenly.
- **Enable Sorting:** When enabled (selected) Enable Sorting enables the user to sort the list (ascending/descending).

Context menu for the resources:

- **Edit Template:** Opens the template in Template Wizard for editing.
- **Find:** Displays the Resource Search.
- **Show Images:** Adds the Image column to the Template list, and shows thumbnails of the scenes in this column.

8.8.2 Working with Templates

Opening a Template

Double-click a template to open it, or right-click and select Open from the context menu.

Selecting a Concept and Variant

1. Open a template from the resource panel.
2. In the template's window, click the Concept and Variant drop-list, and select a concept and variant.

8.8.3   Concepts and Variants
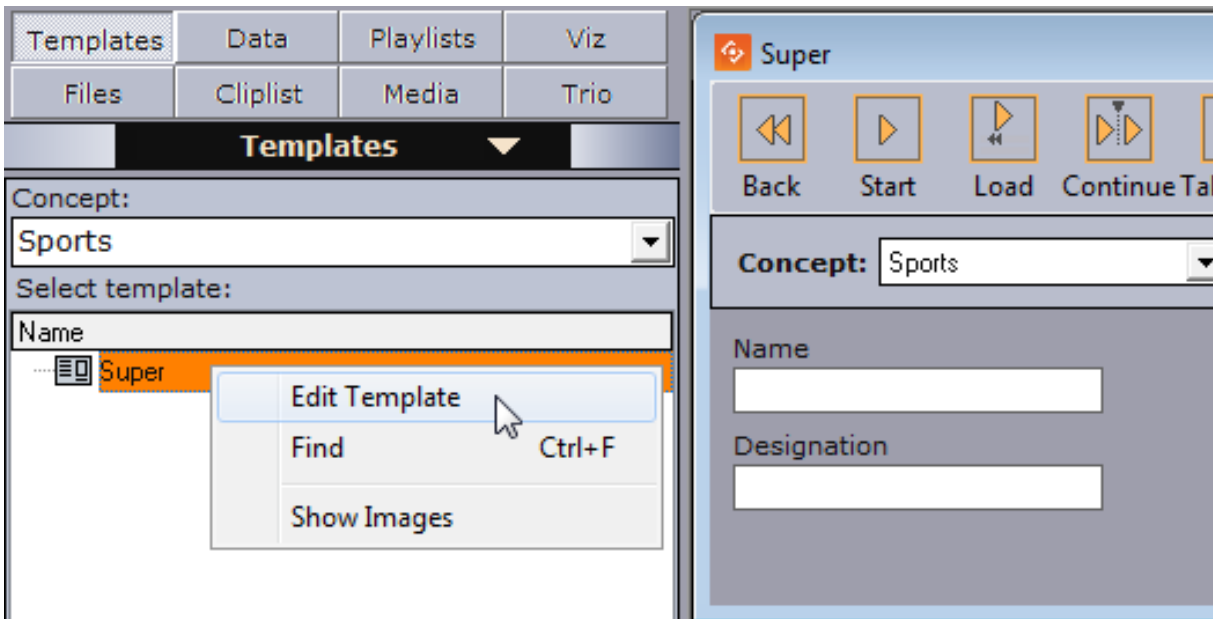
> ✅ **Tip:** The template list is alphabetically ordered (ascending). Click the **Name** column to reverse the order (descending).

The concept drop-down list in the resource panel displays all available concepts, and is used to filter the templates that belong to each concept. When saving a template, the data element will be stored with the selected concept and variant. These can be changed later by the operator when the data element is added to the Playlists or by opening the Data element and selecting a different concept and variant.
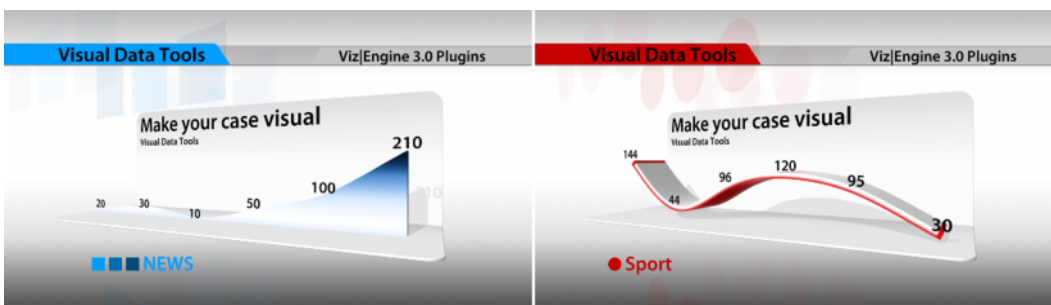
Concepts are a means of reusing content for graphics that have the same properties. Having the same properties enables the template designer to create one template for several scenes that have the same properties. The operator is then able to reuse the parameters saved with the data element and switch concepts and variants without re-entering the data for each Data element in the Playlists.

When using a newsroom system for story creation, the journalist is able to select which concept or variant a specific story is to be aired with. This will save time before a story is put on air, and consequently pushes the deadline for when a story is ready for playout to near real-time.

In addition, the control room operator has the freedom to override these settings in order to reuse any content, as long as the template supports the concept and variant to be used.

> ⚠ **Note:** Concepts and variants are created and organized in Template Wizard. Templates are stored in the Viz Pilot database.

8.8.4   Template Control Buttons

Each template has the same set of control buttons and options:
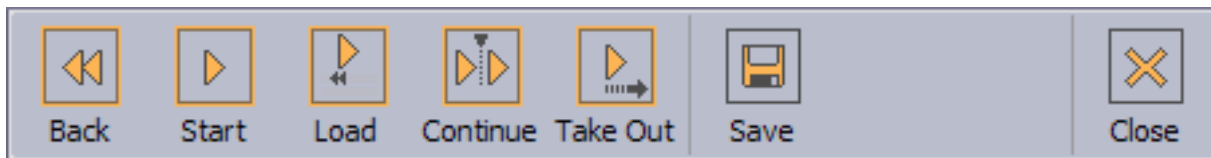
- **Back:** Rewinds the scene to frame 0.
- **Start:** Runs the scene from the current frame.
- **Load:** Loads the scene on the engine that is defined on the active channel.
- **Continue:** Continues the scene from the stop point.
- **Take Out:** Takes the scene to the out state stop point. If the scene is based on transition logic, the scene is cleared.
- **Save:** Opens the save dialog saving the current data added to the template. Will also open the spell checker before the save dialog if spell checking is enabled and words have been misspelled.
- **Close:** Closes the template.

When you open a template, the selected profile will have a defined active channel. There can be multiple profile configurations, but only one profile can be active at a time using the selected program channel.

The template control buttons send instructions through the active program channel, they do not require or use any other configurations.

> ⚠ **Note:** This is an advanced feature that has limitations and is generally used with heavily scripted templates. Whenever possible, templates should be saved as data elements and taken from a playlist to access all the available features.

8.8.5   Template Save Dialog Box
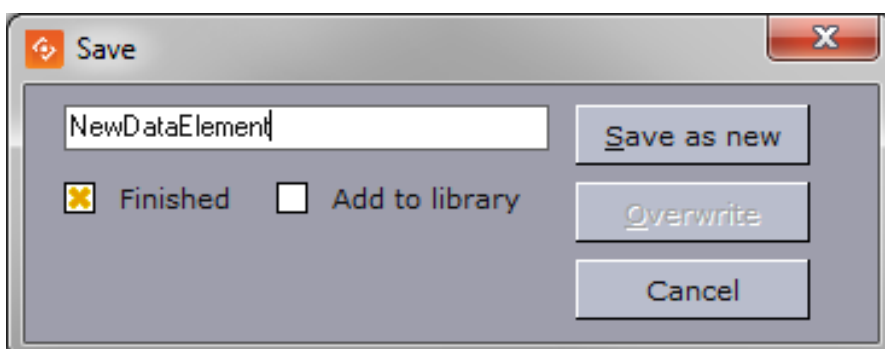
- **Description:** In the text box, type a description that uniquely identifies the data element.

- **Finished:** If the settings are final, select the **Finished** check box. This information will be stored on the database. If a data element is finished it can be shown in the **Finished** column in the playlist displaying either *1* for finished and *0* for not finished.
- **Add to library:** Select this checkbox if the data element will be used often and you want to prevent it from being deleted. The data element will be saved as a *library element*, and all concept and variant information is kept. This prevents the data element from being deleted when using **Options > Delete Data Items**. Library elements can still be deleted manually. If the original template is deleted in Template Wizard, all data elements based on the template will also be deleted, including library elements.
  The **Add to library** checkbox will by default be unchecked. To **Overwrite** an already existing library element, select the **Add to library** checkbox, click **Overwrite** and click **Yes** in the confirmation dialog.

⚠️ **Note:** Data elements that are saved as library elements, will appear with an asterix (*****) after their name in the Data element list.

- **Save as new (ALT + S):** Creates a new data element.
- **Overwrite (ALT + O):** Updates a previously created data element. If the data element is new, the **Overwrite** button will not be available.
- **Cancel:** Cancels the save operation.

8.8.6   Template Spell Checker
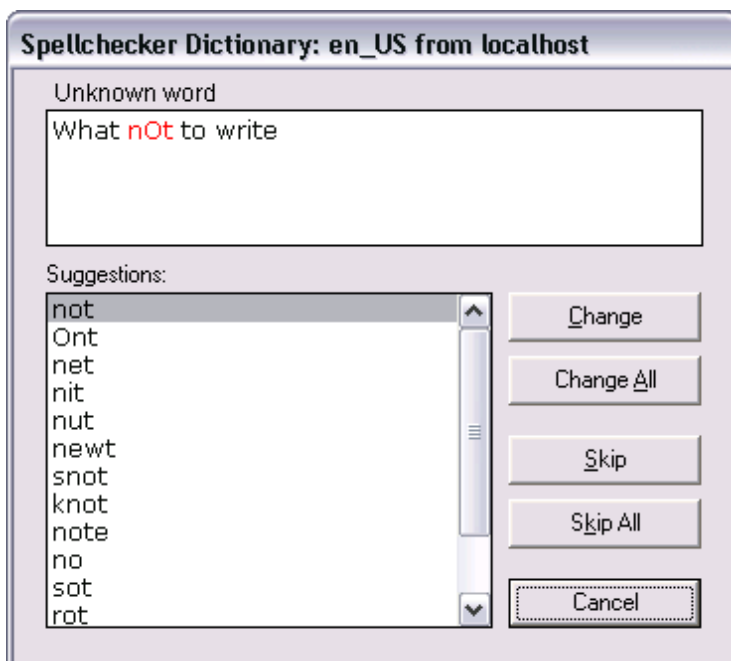
The template spell checker processes the text as part of the save operation. If spell checking is enabled, the spell checker dialog will appear.

Spell checking is enabled by the template designer using Template Wizard. Spell checking can be enabled in Unicode Edit Component and Unicode Memo Component text fields.

- **Change:** Changes the misspelled word (red) with the suggested word before moving on to the next misspelled word. When done, the spell checker opens the save dialog.

- **Change All:** Changes all occurrences of the misspelled word (red) in a single operation before moving on to the next misspelled word. When done, the spell checker opens the save dialog.
- **Skip:** Skips the misspelled word (red) before moving on to the next misspelled word. When done, the spell checker opens the save dialog.
- **Skip All:** Skips all misspelled words and opens the save dialog.
- **Cancel:** Stops the spell checking operation and returns to the template editing view. The save dialog is not opened.

> ⚠️ **Note:** When creating multiple data elements based on the same template, the spell checker will remember previous spelling corrections and automatically apply them to new data elements before they are saved and the template is closed.

8.8.7   Timeline Editor Preview

Timeline Editor Preview in Director

A basic version of the Timeline Editor is available for previewing video elements from within a template or data element. Video preview and scrubbing is available, however the controls for timeline manipulation are disabled.

Context Menu for a Video in a Template or Data Element
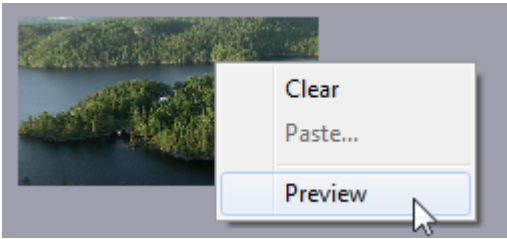
The Preview option is available from the context menu of the video control.

Preview is only available if the image component in the template lists is VideoMediaSearch in the ImageSources for the template, and is only enabled if a video asset is actually selected.

See Template Wizard Components for information on image components.

8.9 Data
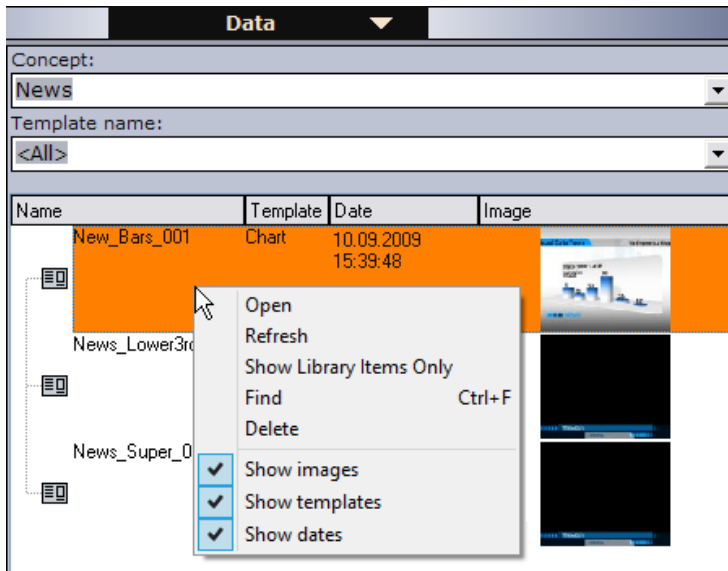
The data elements view displays a list of all data elements or see a filtered view by selecting concept and/or template from the drop-lists.

> ⚠️ **Note:** A concept that is selected in the Templates view is reflected in the Data view, and vice versa.

- Data elements are described with name, template name, date of creation and a thumbnail (if the Thumbnail Generator is used).
- Data elements are based on templates and can be created by the Director operator; however, in most cases data elements are created using Viz Pilot News, for example by journalists. Data elements are then used to populate the playlist.
- Data elements that have been saved as library elements are suffixed with an asterix (*****), see Template Save Dialog Box.

This section contains the following topics:

- Context Menu
    - Columns
    - Resources
- Creating a Data Element
- Editing a Data Element
- Deleting a Data Element

8.9.1   Context Menu

Columns

- **Date:** Shows the date and time the data element was created.
- **Template:** Shows the template the data element is based on.
- **Auto Width:** When enabled (selected), Auto Width expands the columns and distributes them evenly.
- **Enable Sorting:** When enabled (selected), Enable Sorting enables the user to sort the list (ascending/ descending).

Resources

- **Open:** Displays the selected data element in the Work area.
- **Refresh:** Refreshes the elements in the Data list.
- **Show Library Items Only:** Shows only the elements that have been created in previous versions of Director, and that are stored in the library.
- **Find:** Displays the Resource Search.
- **Delete:** Deletes the selected data element. A confirmation dialog box appears when trying to delete library elements.
- **Show images:** Displays the Image column.
- **Show templates:** Displays the Template column.
- **Show dates:** Displays the Date column. This column is displayed by default, and the data elements are sorted by date. Clicking the title area of the Date column reverses the sorting.

8.9.2   Creating a Data Element

1. Open a template from the Templates resource list.
2. Fill the template with data (for example images, text and so on).
3. Click **Save**.
4. Enter a name in the Template Save Dialog Box and click **Save as new**.

> ⚠️ **Note:** Please check *Template Wizard Standard Components* for more details regarding the Unicode Memo Component.

8.9.3   Editing a Data Element

Double-click the data element or right-click it, and select **Open**.

8.9.4   Deleting a Data Element

Select the data element and press the **Delete** key on the keyboard or right-click it, and select **Delete**.

8.10   Viz
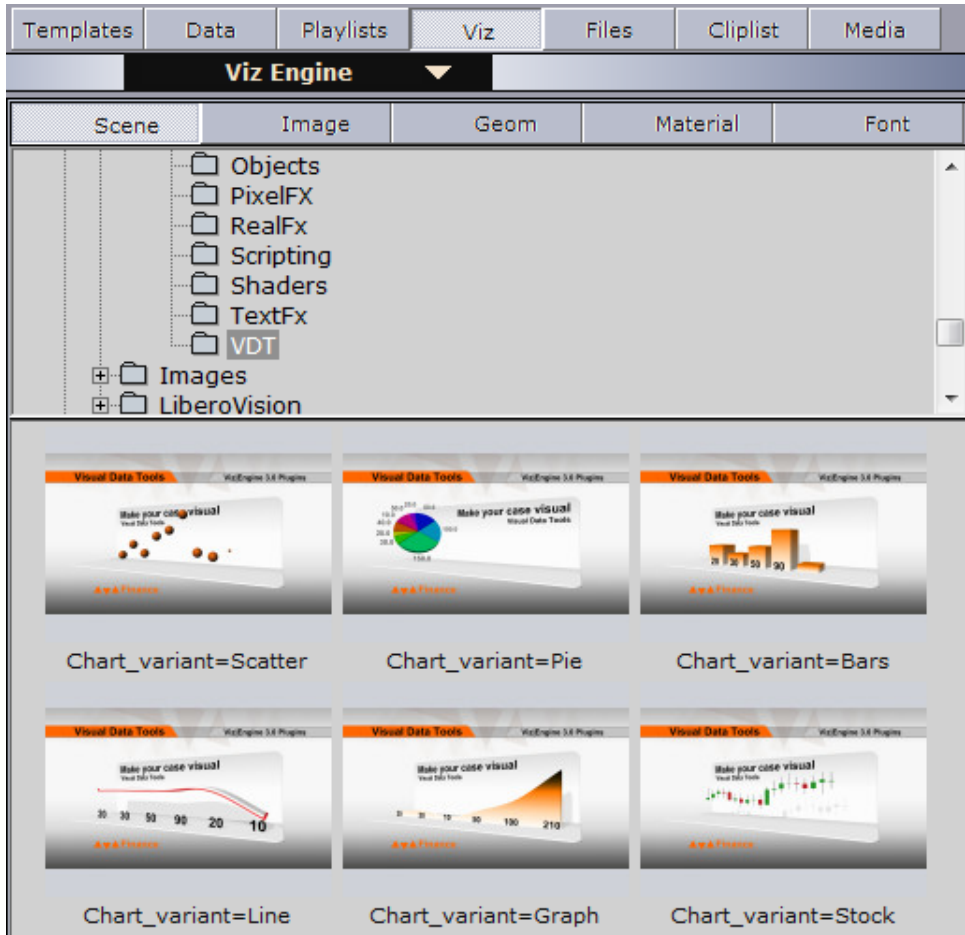
The **Viz** button refers to the currently configured Viz Engine preview channel and its Graphic Hub connection. If you click the **Viz** button, Director shows the different scenes, images, objects, materials and fonts available to Viz Engine.

The Viz panel (also called the *treeview browser*) also has a range of buttons and tabs that display different views of different items and objects available to Viz Engine. The display area shows the directory and the contents of the currently selected folder. Options for refreshing the views and sorting the order of the listed items are described below.

> ⚠ **Note:** Viz images can be played out as fullscreen stillstore images; however, this requires a still store scene. See Working with Stillstore Images.

- **Scene**: Shows available Scenes in Graphic Hub. Server relates to the Server - Scenes option in Viz Artist.
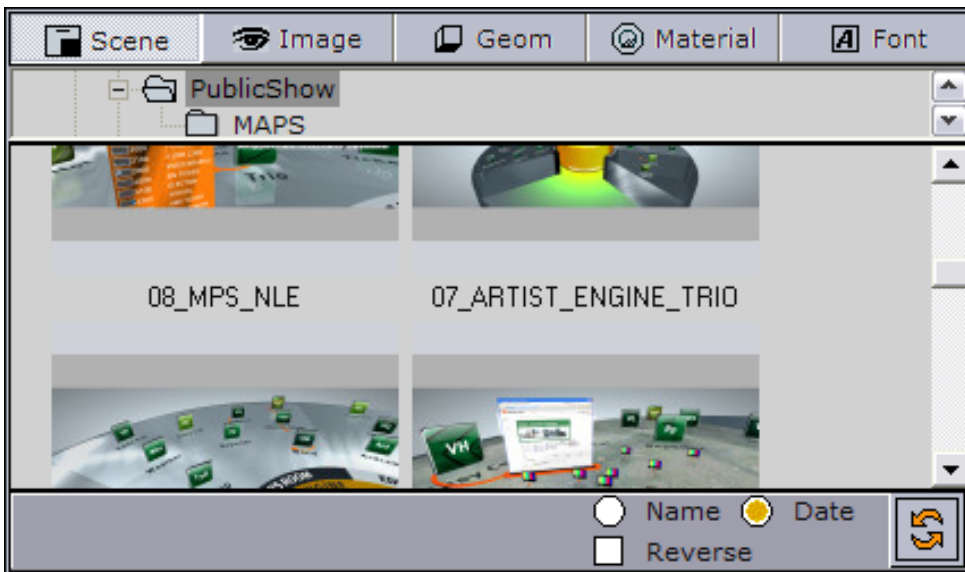
- **Image**: Shows available Images in Graphic Hub. Image relates to the Server - Images option in Viz Artist.
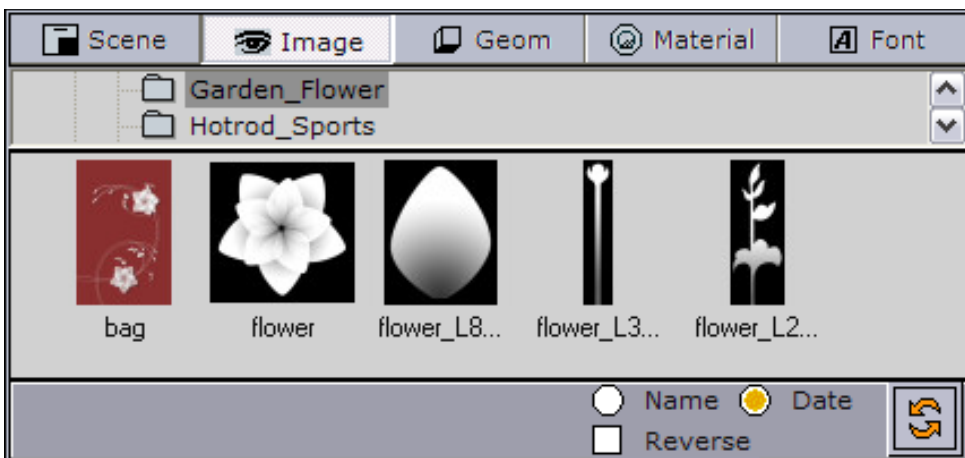
- **Geom**: Shows available Geometry Objects on Graphic Hub. Geom relates to the Server - Objects option in Viz Artist.
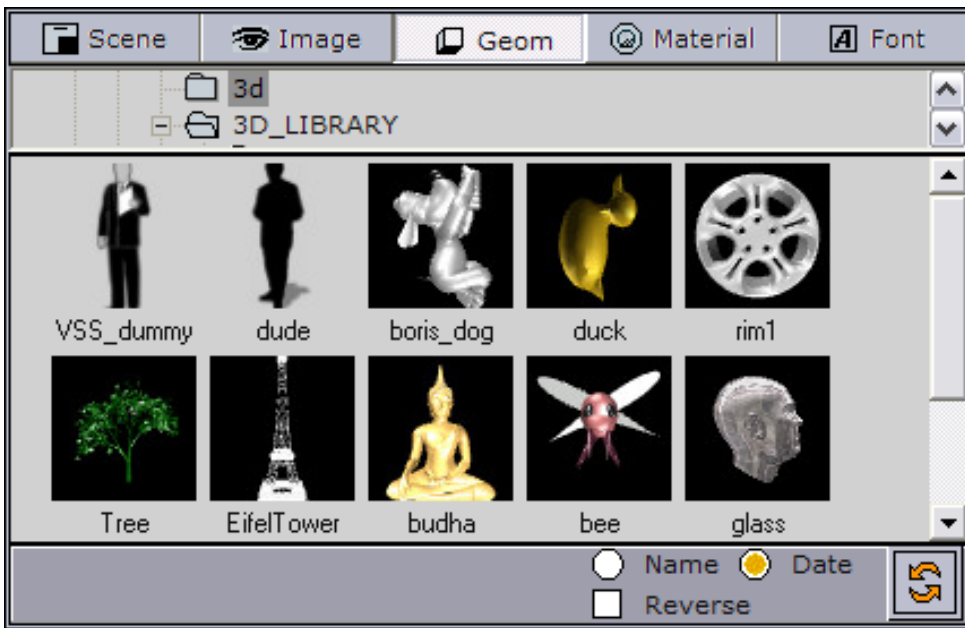
- **Material**: Shows available Material (colors) on Graphic Hub. Material relates to the Server - Material and - Materials Advanced options in Viz Artist.
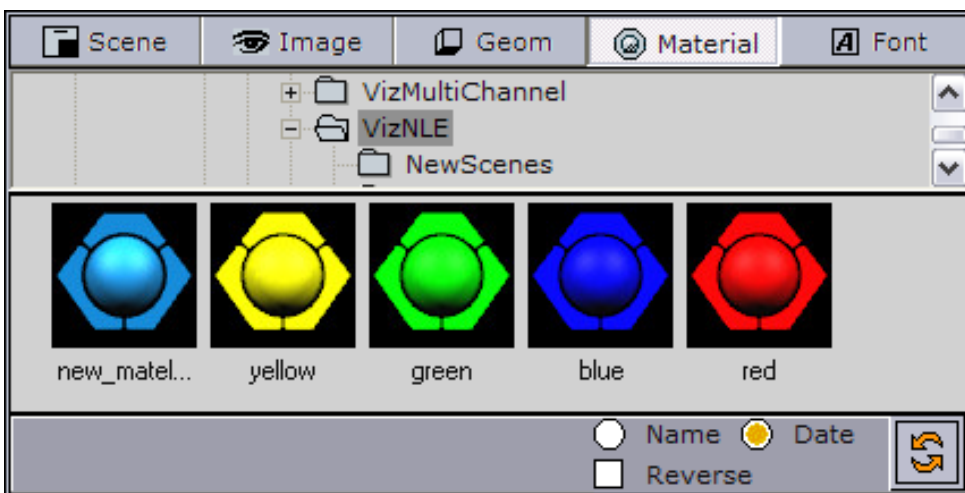
- **Font**: Shows available Fonts in Graphic Hub. Font relates to the Server - Fonts option in Viz Artist.
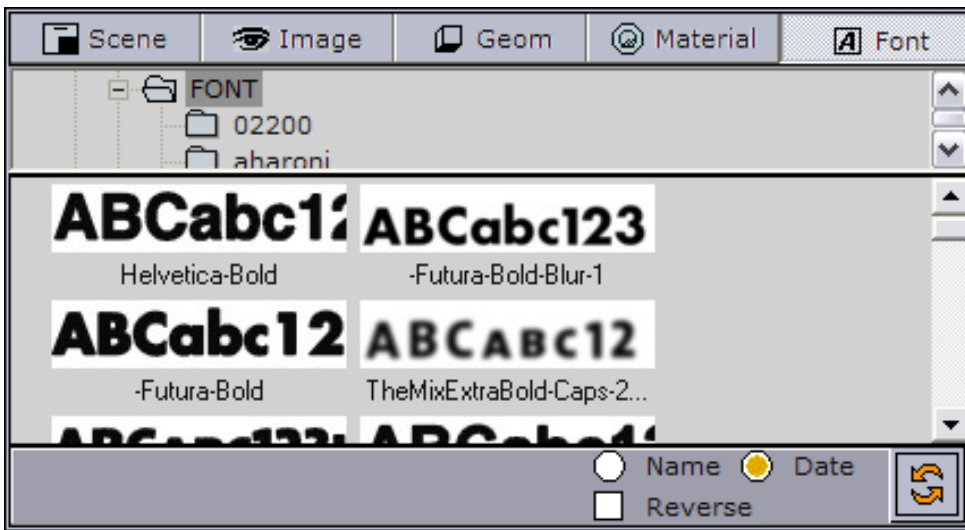
⚠️ **Note:** Geom, Font and Material folders can be placed anywhere on the Graphic Hub.

8.11   Files
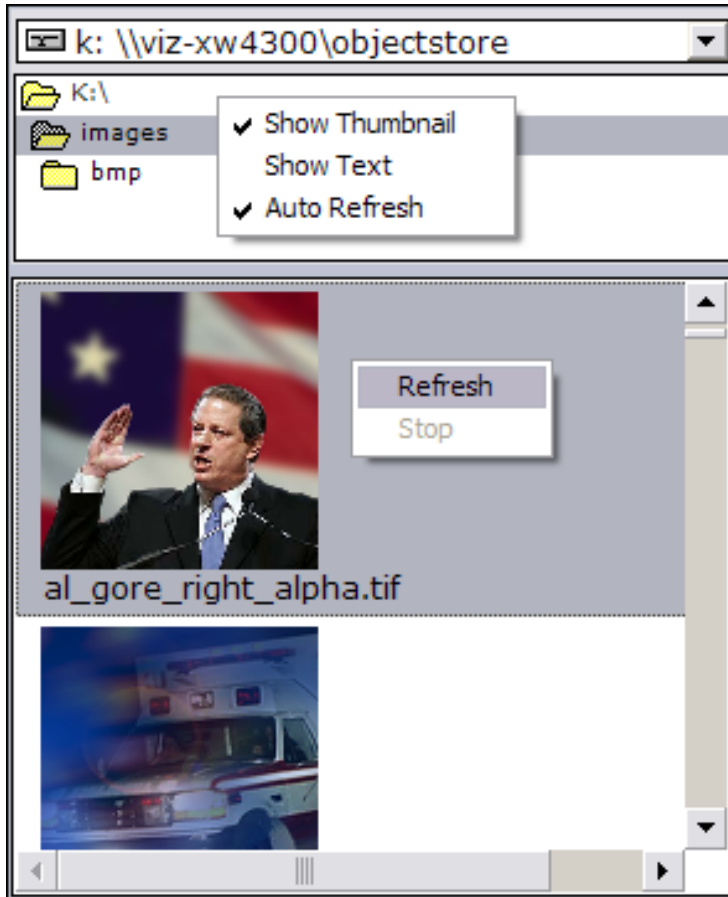
The Files view lets you use external images from local and remote locations. The drop-down list shows the local drive and all drives that are mapped to the computer. The display area shows the external images found for each location. Double-click an image to open a preview of the image.

8.11.1   Context Menu

- **Show Thumbnail:** Displays thumbnail images instead of text.
- **Show Text:** Displays text information relating to pictures instead of thumbnails.
- **Auto Refresh:** Automatically displays the content of the folder selected in the Folder list.

⚠️ **Note:** External images can be played out as fullscreen stillstore images; however, this requires a still store scene or *Image with Name and Title Linking* component. See Working with Stillstore Images.

✅ **Tip:** An alternative to using the Files browser is to use isFileOpen in an image component instead.

8.12   Trio

Viz Trio resources can be used in a Director playlist. In Viz Trio, all shows and pages are stored and organized on the Media Sequencer (like Director's data elements).

This section contains the following topics:

- Trio Panel
- Context Menu

8.12.1   Trio Panel

Show View
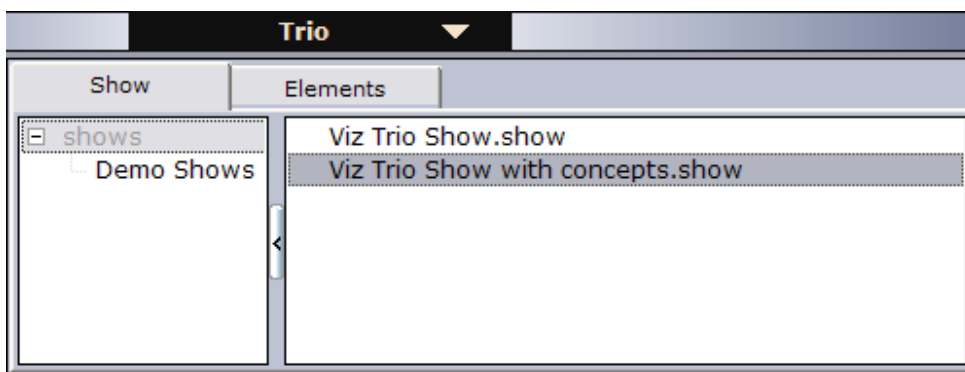
When clicking the **Trio** button, the **Show** tab is selected by default. The left pane displays folders that organize the different shows stored on the Media Sequencer. The right pane lists shows organized under each show folder. A show contains pages (data elements).

> ✅ **Tip:** Hide the Show pane by clicking the **vertical oblong arrow** button.

Elements View
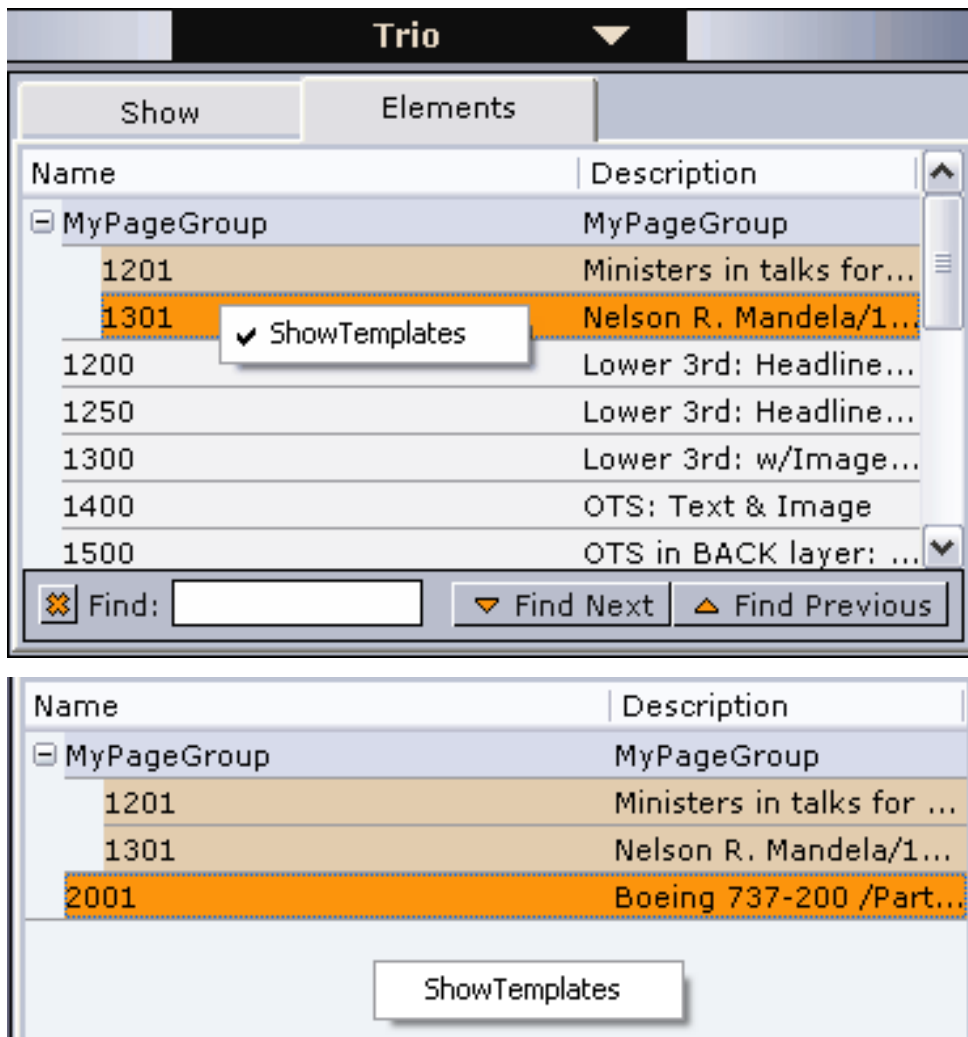
8.12.2 Context Menu

Columns

- **Name:** Shows the names of the elements.
- **Description:** Shows a description of the elements.
- **Auto Width:** When enabled (selected), Auto Width expands the columns and distributes them evenly.
- **Enable Sorting:** When enabled (selected), Enable Sorting enables the user to sort the list (ascending/descending).

Resources

Show Templates displays all templates and all pages in the currently selected show.

Opening a Show

Double-click a show (for example Viz Trio Show.show) to see its pages (data elements).

Adding Elements to a Playlist

Drag and drop pages (data elements) from the Elements view onto the playlist.

8.13   Media

The Media tab lets users search for media assets (still images and video clips) that can be added directly to the story (for example a full screen video clip).

Media information resides on storage systems such as Object Store and Viz One. Object Store traditionally stores still images and person information. Viz One traditionally stores video, audio and video stills. The Media tab combines these sources into one.

This section contains the following topics:

- Media Tab
- Details View
- Search and Filter Options
- Order Management

8.13.1   Media Tab

Media Tab with search results list and Details View:

The top pane shows the Search and Filter Options and a list of categories, while the bottom pane displays the search results and, optionally, the item's metadata.

Search results are limited to a 100 hits per source (for example Object Store or Viz One) in order to reduce network load and to increase the speed of the search. The most recent assets are displayed first. If your results exceed 100 it is recommended to refine the search.

If your search does not find the media you need, you can submit an order if you have Order Management configured.

> ⚠️ **Note:** Depending on the data available from each provider, the status bar will display either *ProviderName=N of M shown* or *ProviderName=N found*, where N will increase as more items are fetched. If a provider does not provide sufficient information, the status bar will display *ProviderName=unknown*.

Context Menu for Search Results
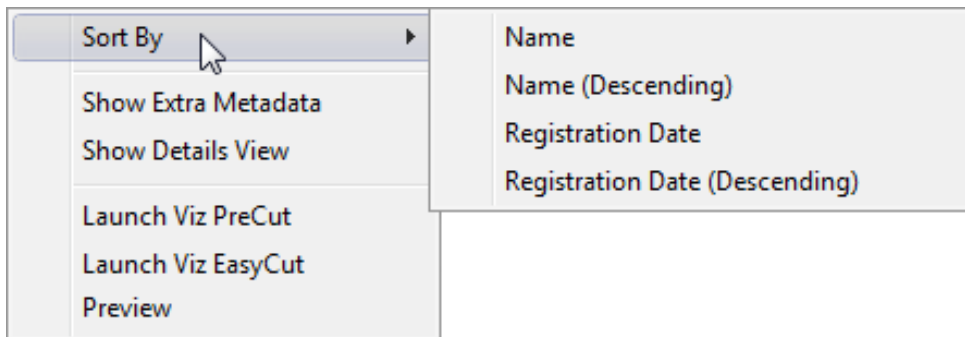
Media Search Results Context Menu

- **Sort By:** Displays a sub menu with sort options.
    - **Name/Name (Descending):** Sorts by name in ascending and descending order.
    - **Registration Date / Registration Date (Descending):** Sorts by registration date in ascending and descending order.
- **Show Extra Metadata:** Switches the media icons to display metadata such as complete filename, creation date, clip length and so on.
- **Show Details View:** Displays metadata of the selected media item in a pane next to the Media pane.
- **Launch Viz PreCut:** For items stored in Viz One, this opens the selected video clip(s) in Viz PreCut for editing.
- **Launch Viz EasyCut:** For items stored in Viz One, this opens the selected video clip(s) in Viz EasyCut for editing.
- **Preview:** Previews images using the Windows Picture and Fax Viewer. Is only available for Object Store items.

Media Search Errors

- **Startup Error:** An error occurred during startup. The host (or Viz Pilot News) will need to be started again to clear the error.
- **Search Error:** An error occurred during a search. Try doing the search again.

8.13.2   Details View

Display an item's metadata by selecting **Show Details View** from the context menu in the search results.

The Details View shows the metadata that is received from the asset search provider. The type of data available can vary, and can be configured in each asset search provider.

The Details View can be placed at the bottom of the search pane or to the right. Use the context menu in the Details View to change the location.
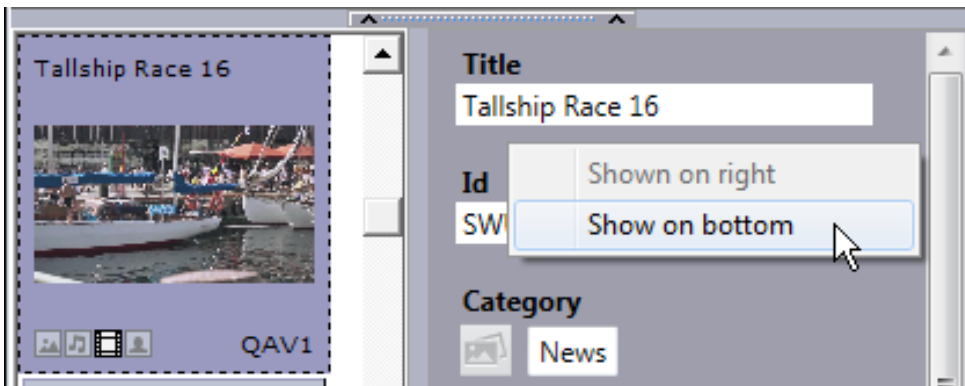
8.13.3   Search and Filter Options
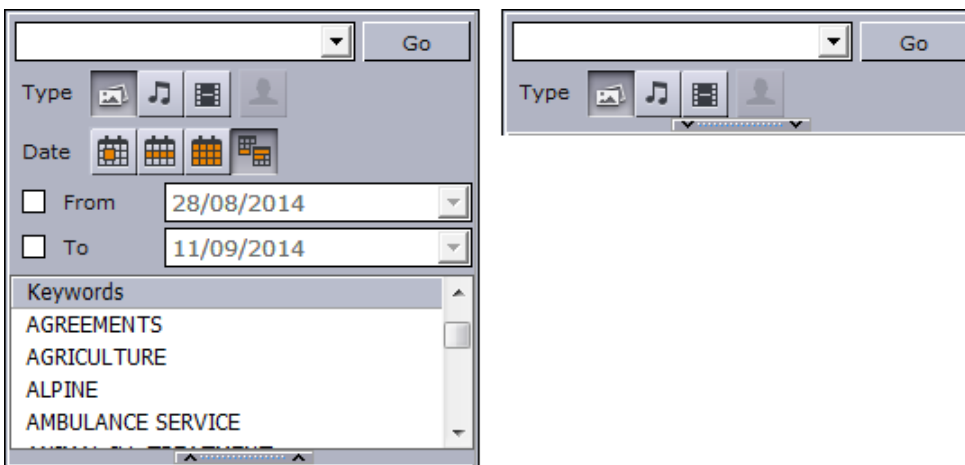
Both Object Store and Viz One searches support free text, keywords, and from and to dates. When the Search box is minimized, only the Search field and Type filters are displayed. Combine the use of filters, text search, and keywords to narrow down your results.

- **Search field:** Combo box for entering search criteria. Previously entered search criteria are remembered per session.
- **Type - Stills, Audio and Video:** Filters the search result displaying only images, audio or video, according to which filters are enabled/disabled. Use **CTRL + click** to select multiple types.

⚠ **Note:** Director does not support the use of audio files.

- **Type - Person:** Filters the search result displaying only pictures that are defined as person (profile) images in Object Store. See also Filter Media by Person Name.

> ⚠️ **Note:** The Person Filter button is only available if a RestVOS Search Providers is configured (see the Pilot Data Server Administrator Guide). If using standard Object Store, search for person name by typing in the Search field.

- **Date:** Filter results based on date range. Select from **Last 24 hours, Last week, Last month, Custom date range**.
  Filters the search result based on registration dates. For Viz One searches, the default date used to search is the CreationDate (the date the item was created in the Viz One), however, this can be configured to use other dates from the metadata (for example news.eventDate, asset.recordingDate, asset.publicationDate, asset.retentionDate).
- **From/To:** When the **Custom date range** button is selected, the From and To dates are displayed.
  A red background indicates that the from-date specified is later than the to-date.
- **Keywords:** Filters the search result based on keywords. Templates that use the RequiredKeyWords property will have the relevant keywords pre-selected in the list. Keywords are configured for each Service Provider. See Required Keywords.
  Note that in Director, the Media Search is not bound to a specific template or data element, which means that the search results cannot be limited to what's relevant for a specific component in a template. This means that all images show up in the search, but you are not allowed to drop them on the image component if they do not respect to the keywords filter.
  Still store images can be played out as fullscreen stillstore images; however, this requires a still store scene. See Working with Stillstore Images.

If multiple Viz One systems are in use, all of the systems will be searched and the results will be combined according to the ID of the clip - if the same clip is available on multiple Viz One systems, it will only appear once in the search results.

> ⚠️ **Note:** If your search does not find the media you need, you can submit an order if you have Order Management configured.

Drag and Drop Media

When selecting media assets from a search, both images and videos can be added to playlists, templates and data elements using drag-and-drop. Drag selected media to the receiving media container. The media asset thumbnail will then replace the placeholder icon in templates and data elements.

In addition to resources from media searches in Viz Pilot, external applications can be programmed to be a provider of media resources using Drag and Drop. The required API for this is specified in the Appendix, see Integrating Drag and Drop from External Applications.

8.13.4   Order Management

The Order Management System is designed to close the gap between newsroom users ordering content and the other department in the building fulfilling these requests. The Order Management System is built on top of Viz One, Vizrt's media asset management solution.

For more details, see Order Management and the **To configure Order Management in the Setting Parameters** section in the Pilot Data Server Administrator Guide.

Submitting a Work Order from Director

Follow the procedure Submit a Work Order from Viz Pilot News.
The only difference is that to begin submitting a work order, you *drag* the Order icon to the image component in the template, rather than clicking an Order button.

8.14   Playlist Window

The playlist window in Director is used to add items (playlist elements such as full-screen images, videos, template based scenes and so on) and to select how items are ordered, grouped and played out.

New items can be added to the playlist in Director, or they can be added from external applications such as Newsroom systems. The playlist and its elements can be configured in the playlist window.

A new playlist is created by clicking **File > New Playlist** or by the context-menu (right-click) in the playlists overview window in Director. Each playlist can be separately Activated or Deactivated as required.

This section contains:

- Organizing the Playlist
- MOS Playlists
- Using the Playlist
- Playlist Areas

8.14.1   Organizing the Playlist

Use the shortcut menu in the playlist columns described below to organize a playlist.

Playlist Columns

After opening the playlist window (see Playlist Areas) right-click a column header to expose a shortcut menu with options for hiding and showing all columns in a template. The **Description** column header is right-clicked in the screenshot below.
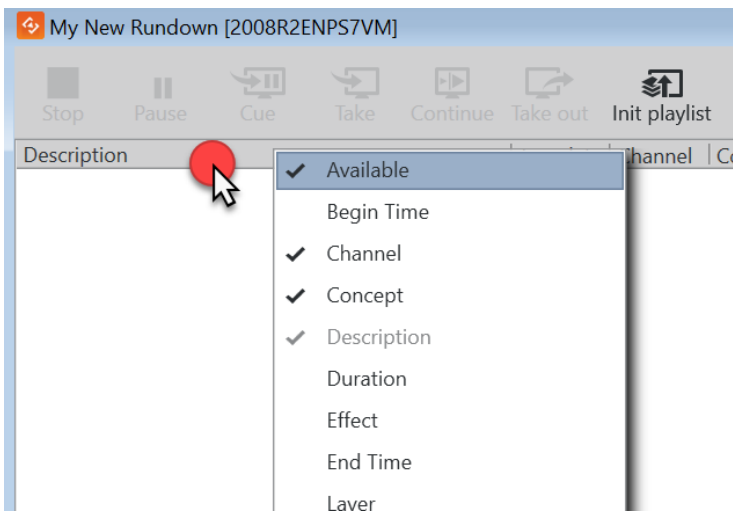
> ✅ **Tip:** Right-click items in the shortcut menu to select which columns should be displayed. Drag the columns to rearrange them.

- **Available:** Displays the status of external resources needed by the Viz Engine (for example transferred video, and if it is available on the video playout engines). Errors are shown as tooltips.
- **Begin Time:** Shows the activation start time for a group (format hh:mm:ss).
- **Channel:** Shows which output channel an element is to be sent to. Various elements can be sent to different output channels. The output channels can be set directly in the column. By default the main [PROGRAM] output channel is selected, but this can be changed by setting an alternative channel for this element only or in a template. If it is changed, all data elements made from the template will then have this channel. By creating a group and placing elements within it, all elements in the group will be organized by having the same channel. The Channel column is presented by default.
- **Concept:** Shows which concepts the playlist element is associated with.
- **Description:** Shows the description of the element. By default this will show the path of the scene (or the element name of a stand-alone media file). May be edited in-line in the playlist.
- **Duration:** Shows the length of the element.
- **Effect:** Opens the Choose Effect dialog, which makes it possible to select a transition effect between two pages.
- **End Time:** Shows the activation end time for a group (format hh:mm:ss).

- **Layer:** Allows loading of graphics in separate layers on Viz Engine (front, middle, back). For example, a lower third can be shown in front of a virtual studio set or any other background, or a bug can be shown in the front layer while a lower third is shown in the middle layer. This column is presented by default.
- **Loaded:** Shows the loaded status (in memory) of the scene and images used for a playlist element of that scene. Errors are shown as tooltips.
- **Loop:** Displays a loop information column.
- **Mark In / Mark Out** : Sets mark in/out times for video clips.
- **MOS ObjAir:** Two states are possible: READY or NOT READY, as defined by the MOS protocol (see www.mosprotocol.com). If READY there is no change to default; if an element or group is marked NOT READY, it will be skipped during playout.
- **Playout Mode:** The user can specify how groups should be played out and can choose between None, Duration and Timecode. See Using the Playout Mode for more information.
- **Story #:** Shows the story number for stories in MOS playlists. This is only supported from the ENPS newsroom system.
- **Template Description:** Shows the template description (for example name).
- **Template Id:** Shows the template ID.
- **Timecode:** The timecode is an offset time in format hh:mm:ss:ff. The timecode indicates that an element should be played out relative to the parent group or video. This is for instance used in composite groups with a video and overlay graphics that is played out on a timeline.
- **Variant:** Select a concept's variant from the drop-list (see the **Concept** column).

> ⚠ **Note:** The last four items on the playlist window shortcut menu allow you to set default playlist layout, among other things.

> ⚠ **Note:** Unlike the other options in the menu, selecting/de-selecting columns does not hide or show the columns in the playlist window.
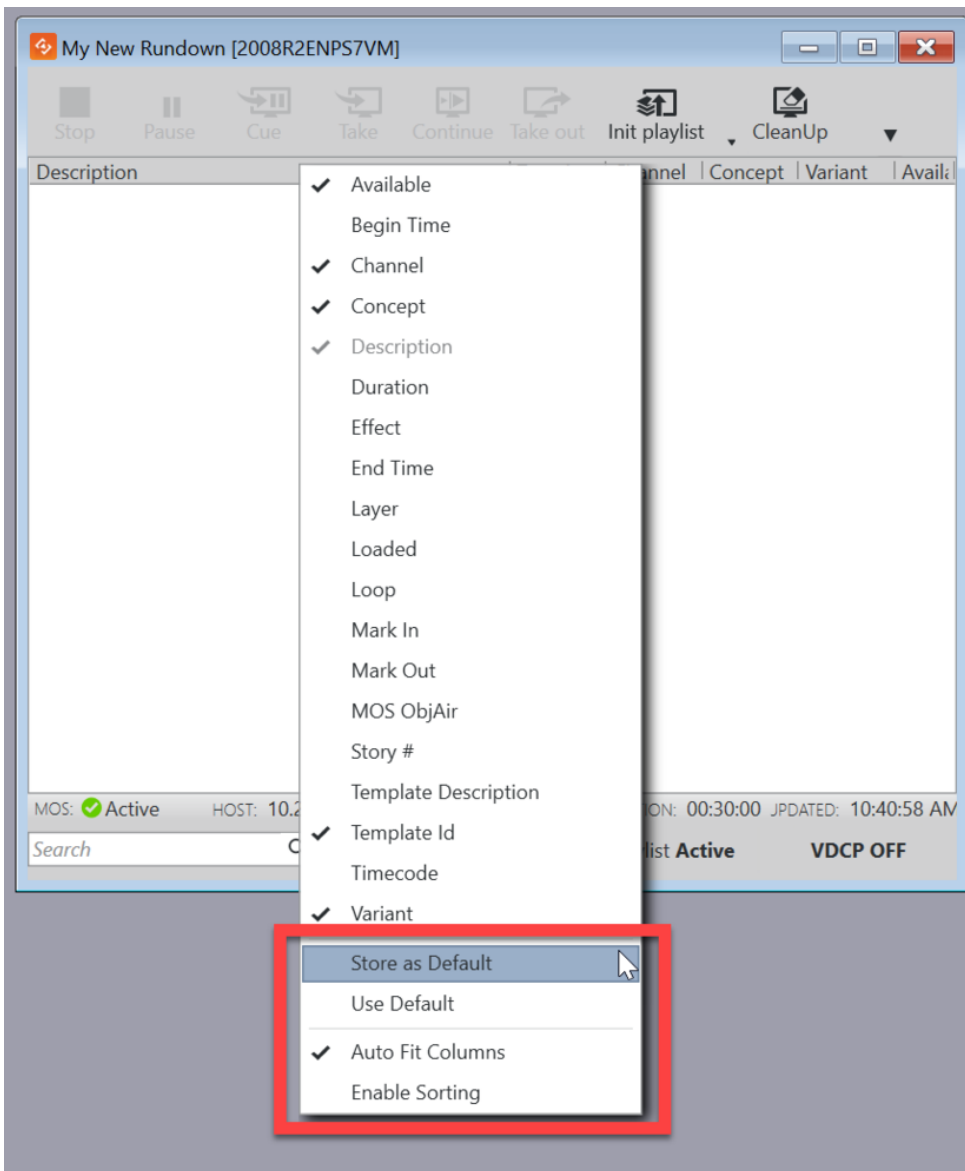
- **Store as Default:** Select this option to use layout and Playlist Pane settings for all new playlists.
- **Use Default:** Reverts to the default layout.
- **Auto Fit Columns:** Automatically fits all displayed columns to the given width of the playlist.
- **Enable Sorting:** Enables sorting. Click a column header to sort items. Items cannot be rearranged by dragging.

8.14.2   MOS Playlists

MOS playlists (newsroom playlists) can be opened in Director once the Viz Gateway Connection has been configured in **Tools > Media Sequencer Configuration >** MOS Tab.

An additional MOS status bar is displayed at the bottom of the playlist window when a MOS playlist is open.

Once Director requests the monitoring of a MOS playlist, it's presented as a playlist. Director then monitors the playlist and receives updates from the newsroom system. The graphics can then be taken on air from Director.
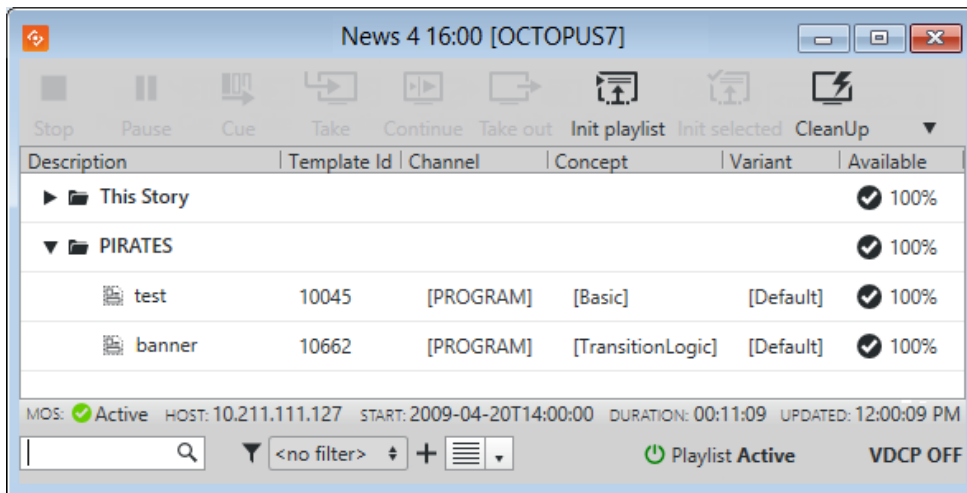
Request or Release a MOS Playlist

When a MOS playlist is first opened in Director, it is empty. Monitoring must be **requested** in order to populate the playlist.

Right-click on the MOS Status Bar to request or release a MOS playlist:

- **Request:** Begin monitoring, and populate the Director playlist with the elements from the newsroom system. Once the monitoring is active, all subsequent updates done to the newsroom rundown will be reflected in the active Director playlist.
- **Release:** Director will stop monitoring the newsroom system. A list of elements will still appear in the Director playlist, but any subsequent updates to the Newsroom rundown will not be reflected in the inactive Director playlist.

> ❌ **IMPORTANT!** Any elements that are manually added to the playlist by Director will disappear if a released playlist is requested again.

> ⚠ **Note:** If using Director to take items on air, be aware that the cursor position in Director may change if the related item is deleted from the newsroom rundown. The Current Element cursor would move to the following item, and the Next Element cursor would be removed.

MOS Status Bar

The rundown monitor bar displays the following:

- **MOS:** The current MOS status for the playlist (Active/Inactive).
- **Host:** IP address of the Media Sequencer that has requested the playlist.
- **Start:** Playlist start time, according to the Newsroom system.
- **Duration:** Playlist duration, according to the Newsroom system.
- **Updated:** Last update time.
  CleanUp and **Auto Preview** work on playlists that are either active or inactive. **Initialize** works once the MOS playlist has been populated with content.

8.14.3   Using the Playlist

This section covers the following topics:

- Playlist Pane
- Playlist Cursors
- Playlist Tool Bar
- Using Playout Mode
- Playlist Filter and Filter Editor
- Playlist Search
- Thumbnail View
- Playlist Status

Playlist Pane

When the playlist is active, you can double-click an item to bring up the template/form where you can fill out information and/or take the item to air. Drag items to rearrange them or add them to groups.

Right-click an item to bring up the playlist shortcut menu:
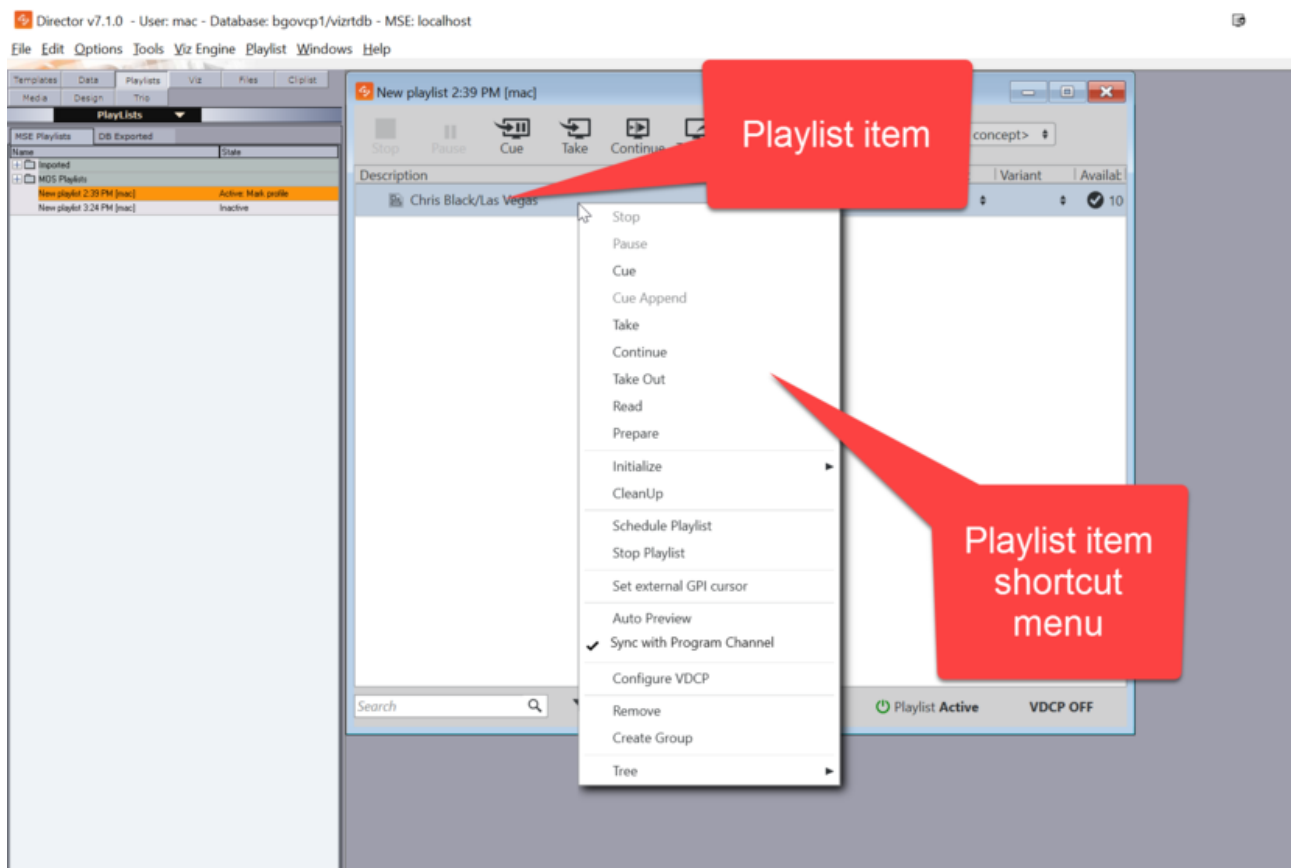
- **Stop** or **Pause** the video.
- **Cue:** Prepares the clip for playout so the first frame is ready in the player.
- **Cue Append:** Prepares the clip for playout. The clip will start automatically when the current clip ends.

- **Take:** Performs a direct take on the selected page.
- **Continue:** Continues the playout of an element.
- **Take Out:** Takes the element out (hard cut).
- **Read:** Reads the element.
- **Prepare:** Prepares the clip in the pending player without affecting the current clip, so it is ready to be played out. The clip will not start automatically after the current clip. Requires Viz Engine 3.3 or later.
- **Initialize Playlist** or **Initialize Element**.
- **Cleanup:** Cleans up all renderers that have an item in the current playlist by clearing all loaded graphics from memory for those channels. To clear all renderers, go to the **Main Menu >** Viz Engine Menu > **Cleanup Renderers**
- **Schedule Playlist:** Activates the playlist. For example, if Begin time has been set, the playlist must be scheduled before the changes take effect.
- **Stop Playlist:** Stops a previously scheduled playlist.
- **Set external GPI cursor:** Sets the external cursor for a selected element. The cursor indicates the element that will be taken if a "take and advance" General Purpose Input (GPI) action is executed, for example.
- **Auto Preview:** Automatically enables preview of next element from selected.
- **Sync with Program Channel:** Syncs the preview with what's showing on the program channel. For example, if you have a lower third on the preview and are showing a fullscreen video on the program channel you can see them together in the preview.
- **Configure VDCP:** Opens the Video Disk Control Protocol (VDCP Tab).
- **Remove:** Deletes selected items.
- **Create Group:** Creates a new group. The new group will be created above the currently selected item. Groups cannot be created when multiple items are selected.
- **Tree:** Organizes the elements in the current playlist and changes the font used.

> ✅ **Tip:** Right-click the column headers and select **Store as Default** to use these settings and to set layout preferences for all new playlists. See Playlist Columns for more information.

Playlist Cursors

A cursor visibly indicates any item being acted on in a playlist. When a playlist is active and playing, the current item playing will be indicated by a **green arrow**.

When GPI is enabled, the external GPI system's cursor will be displayed in any client that is using the same profile as the external system. The **red arrow** indicates which item is next in the playlist according to the external automation system.

Playlist Tool Bar
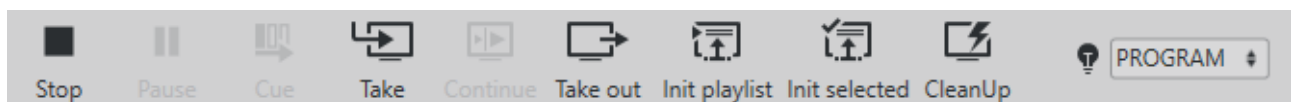
- **Stop** video, if running. Icon active (dark black) if video running.
- **Pause** video, if running. Icon active (dark black) if video running.
- **Cue:** Prepares the clip for playout so the first frame is ready in the player.
- **Direct Take:** Performs a direct take on the selected page.

- **Continue:** Continues the animation on the selected page, or any that is loaded in the same transition logic layer as the selected page.
- **Take out** the selected page, or any that is loaded in the same transition logic layer as the selected page.
- **Init (initialize) playlist:** Loads the playlist on the program and preview renderer.
- **Init (initialize) selected:** Loads selected pages on the program and preview renderer.
- **CleanUp** on the Viz Engine renderer (both visible and loaded memory content).
- **Concept:** Selects the required Concept for the playlist.

Using Playout Mode

Playout Mode can be selected as a playlist column. Playout Mode specifies how groups should be played out and the different options are available using the Playout Mode column drop-down menu. The user can choose between three options: **None**, **Duration** and **Timecode**.

- **Timecode** is the default option. The time is set in the **Timecode** column. Timecode indicates that an element should be played out relative to the parent group or video.
- When **Duration** is selected, elements of the group will be played out one after the other, each staying on air for the amount of time that is specified in the Duration column. It is possible to set a default duration for items without an explicit duration at the bottom of the Playout Mode drop-down menu. The duration of items using this default value will be shown in italics and grayed out in the duration column. However, they can still be edited by first selecting the item row and then clicking the duration value.
  Default duration fields will not appear for the other Playout Mode options.
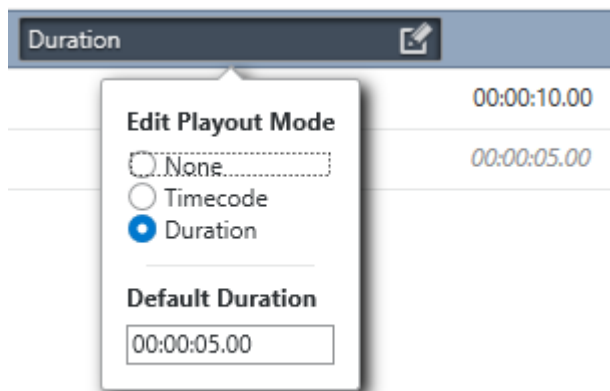
> ⚠ **Note:** The default duration functionality is supported using Media Sequencer 5.0 and later.

> ℹ **Info:** Depending on the selected playout mode for the containing group, the duration column will show one of two different underlying values for each element, internally called *duration* and *take_duration*. When playout mode is Duration, the column will show the value called *duration*, otherwise *take_duration*. However, regardless of the playout mode, both values might exist in the Media Sequencer at the same time; for example, if set in an external application or a previous version of Director. In such cases, *duration* will only have an effect if the playout mode is Duration, while *take_duration* is always considered when taking an element. Regardless of how the element is taken on air, it will be taken out after that time if it has take_duration set. Since the interface only shows one of these values, having both set might lead to

unexpected results. Such conflicts must therefore be resolved by clicking the **Resolve** button that is shown in this state. This will remove the instance of *duration* and *take_duration* that is not currently visible on the interface.
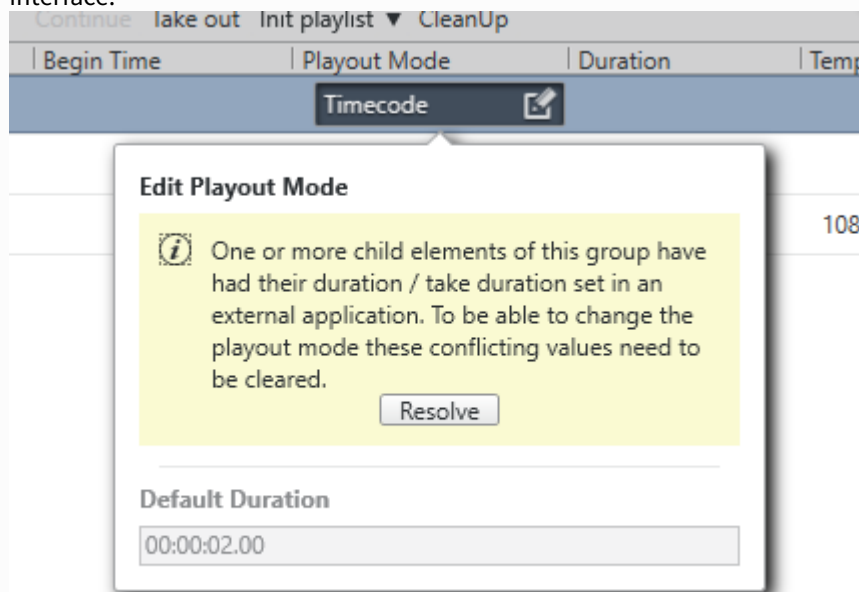
To enable looping of a group, the Playout Mode must be set to **Duration** before the Loop column is toggled on. The duration must be set on all elements in the Duration column. Remember to use the **Stop** button in the Playlist Tool bar in order to stop the loop.

- The **None** option means that there will be no playout.
- There are other options besides None, Duration and Timecode that may be shown in this column. These are the result of playout modes set in other applications that are not recognized by Director. To help avoid accidental changes that cannot be undone through the interface, Director does not allow the playout mode to be changed while such an unknown mode is active.

⚠ **Note:** Nested carousels (groups with playout mode of Duration) are not supported.

Playlist Filter and Filter Editor
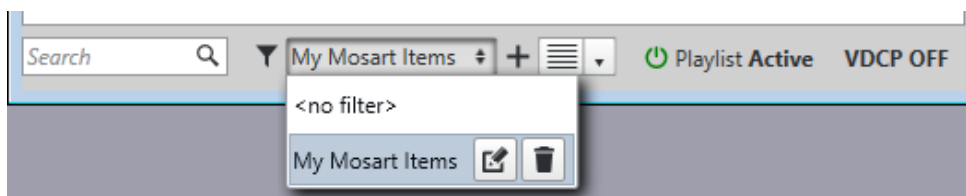
Use the Filter option to filter what items the playlist will show. You can create as many filters as required with the New option, and existing filters can be edited or deleted. When you have created a filter, simply select the filter to use in the drop-down list. Any items that do not satisfy the filter selected will be hidden from display, and not deleted. Deselect filtering to show all items.

In the example below a filter is created that only shows items in the playlist where the description contains the word **Mosart:**

Use the drop-down entries and check-boxes to create a filter as required, based on the template attributes.

> ⚠️ **Note:** To filter by the category assigned to the template, you must select Subcategory in the drop-down menu. This selection will list all categories created using the Template Wizard client to help filter data elements assigned to a given category. See also Categories and Channels.

Playlist Search

Search the playlist. This is a simple search, not a filter. Go to the next match by pressing Enter.

Thumbnail View

Change the thumbnail size.
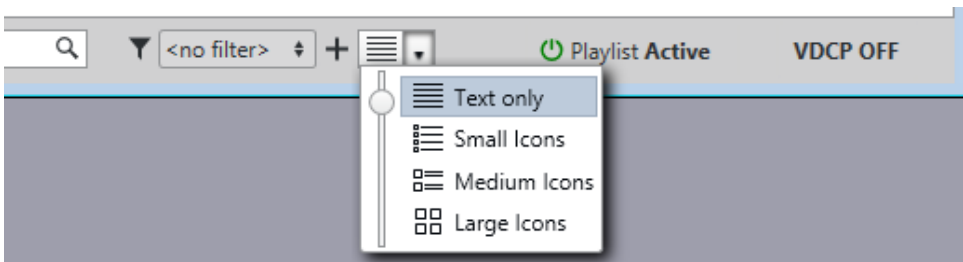
Playlist Status

If there is a VDCP handler for the playlist in the Media Sequencer, this button will display the status. Click the button to set the playlist to active/inactive.

8.14.4 Playlist Areas

Opening a Playlist

1. Start Director.
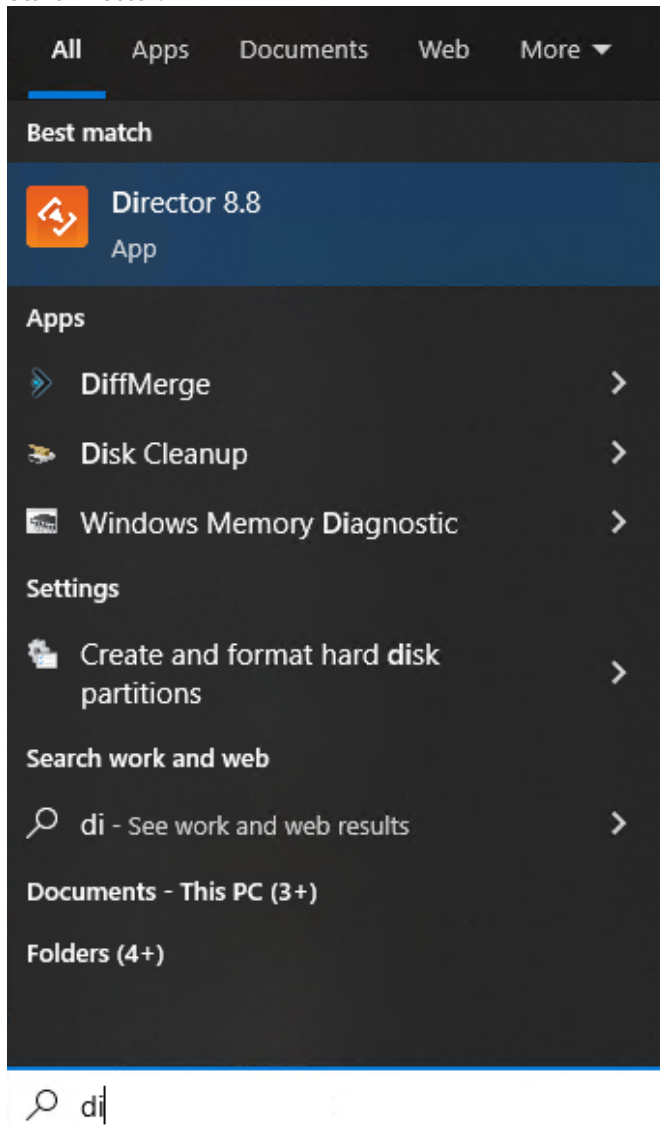
2. Click the Playlists Panel.
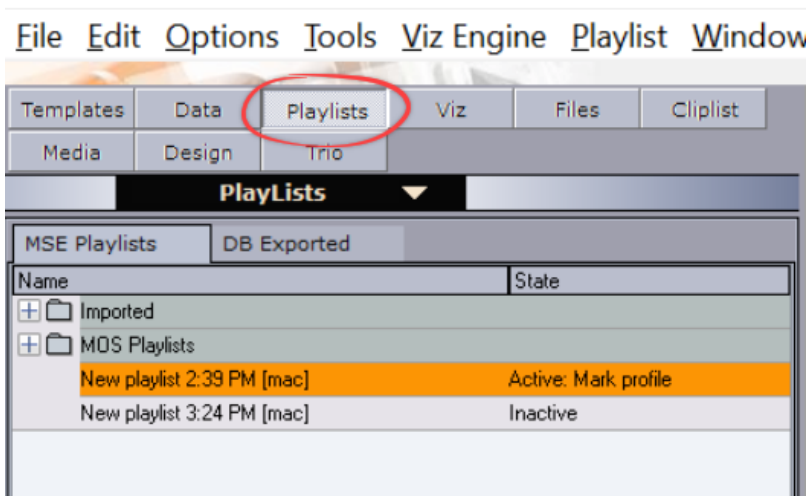
3. Double-click a playlist item.
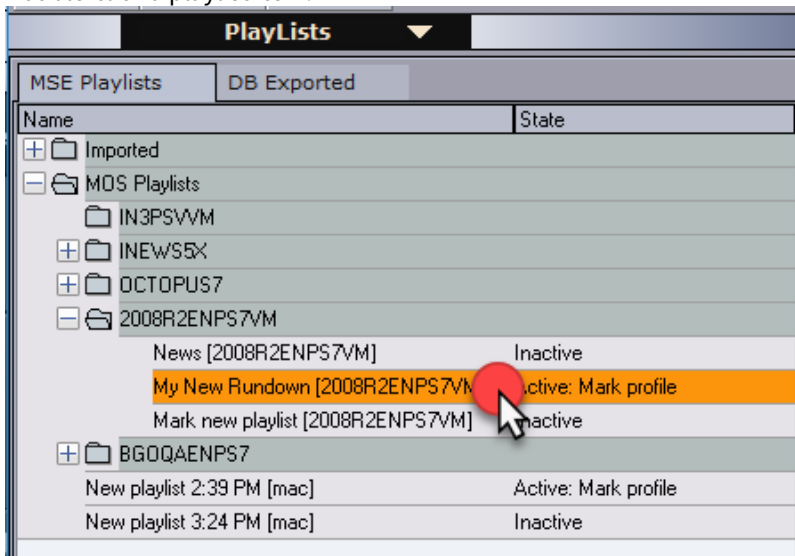
4. The item opens in the playlist window.
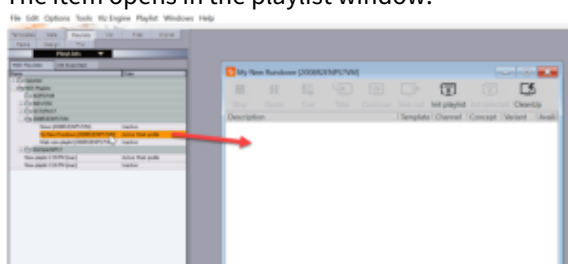
> ⚠ **Note:** In Viz Pilot 6.0 and later, the Playlist Window described is the default. You can select the old Legacy Playlist in the Playlist - General Preferences options.

> ⚠ **Note:** Live Update of templates from Template Wizard has been superseded by Update Services. If you need to use Live Update, you must also use the Legacy Playlist.

Playlist Areas
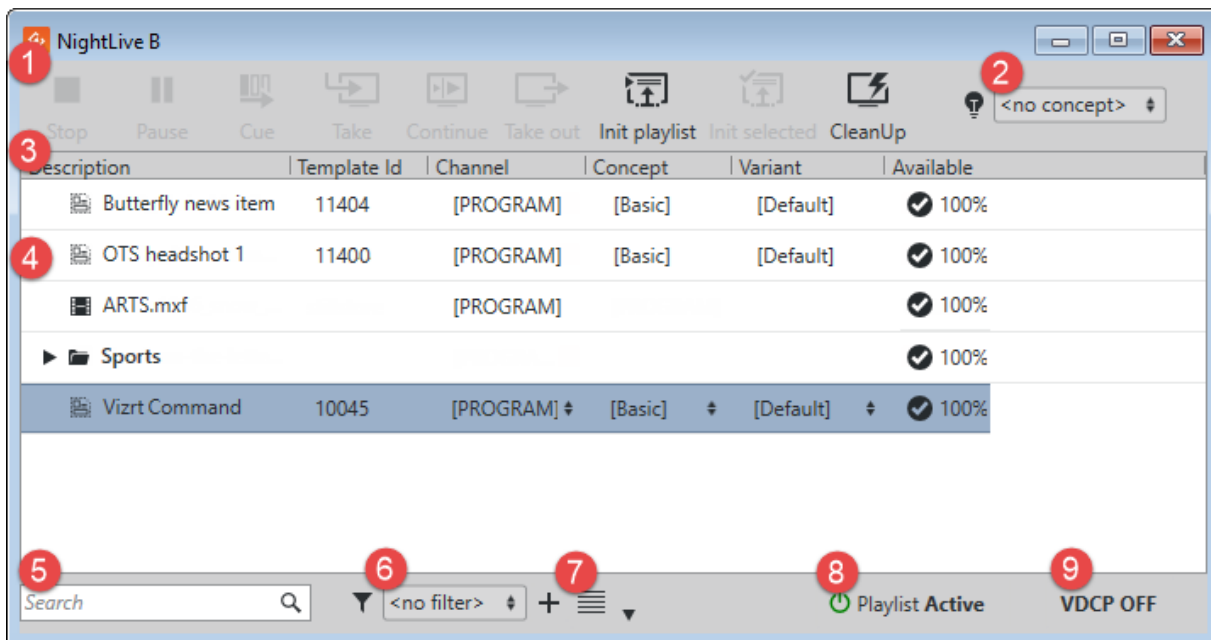
1. Playlist Tool bar.
2. Concept selection.
3. Playlist Columns.
4. The Playlist Pane displays all items and groups in the playlist. See also Using the Playlist.
5. Playlist Search field.
6. Playlist Filter and Filter Editor.
7. Thumbnail View.
8. Playlist Status indicates the current playlist status. Clicking it activates/deactivates the playlist.
9. The VDCP button indicates the current VDCP status. Clicking it opens the Video Disk Control Protocol (VDCP Tab) where you can configure VDCP parameters as well as stop or start the VDCP handler.

⚠ **Note:** An additional status bar is also displayed at the bottom of the window for MOS Playlists.

8.15 Cliplist
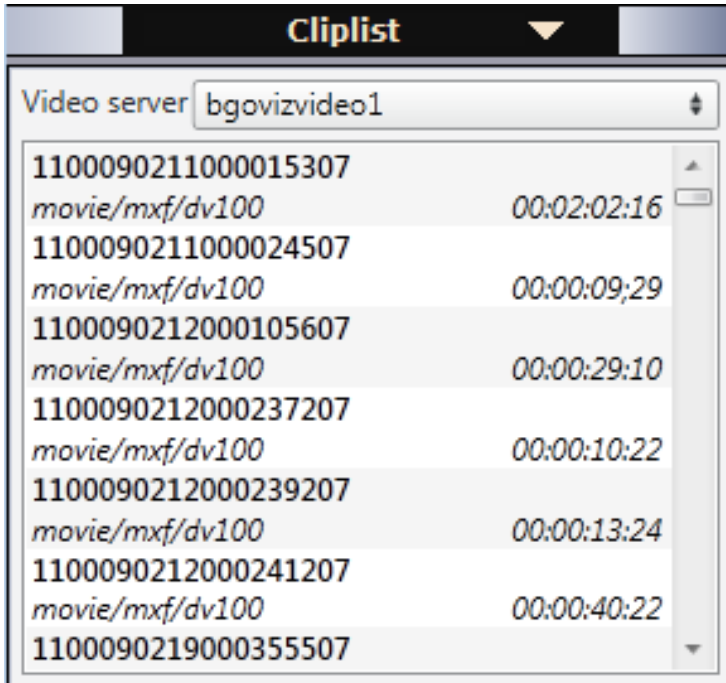
Director can monitor clips on a video device that supports the Multiport Video Control Protocol - MVCP. With MVCP support, operators can add clips to any playlist, trigger the clips manually for playout, and monitor playout status using the Timecode Monitor. The Timecode Monitor displays the timecode, video server channels and clip duration. Video servers are added using the Profile Configuration.

8.16   Playlists

The Playlists panel displays the available playlists on the Media Sequencer.

This section contains the following topics:

8.16.1   Playlists Panel

MSE Playlists Tab

Media Sequencer playlists can be created in two ways:

1. Create playlists in Director by right-clicking inside the MSE Playlists tab and selecting **New Playlist**.
2. In most cases, playlists are created by newsroom systems that are monitored and played out by Director or other control applications such as Viz Trio or Viz Multichannel. These appear in the folder **MOS Playlists**. The newsroom playlists are stored on the Media Sequencer, and updated through the Viz Gateway or Viz Device Manager. Viz Gateway supports most newsroom systems that support the MOS protocol. Viz Device Manager only supports Avid iNEWS Control Air.
MOS Playlists can be opened in Director once the Viz Gateway Connection has been configured in Tools > Media Sequencer Configuration > MOS Tab.

Options

Right click on the column headers in the MSE Playlists tab to select which columns to view (**Created, Modified, Name, Profile, State**).

Right-click in the list to access:

- **New Playlist**, **New Folder**
- **Open**, **Open in New Window**
- **Activate in current profile** / **Deactivate**

> ⚠ **Note:** When the state is set to active and the playlist contains video clips, Media Sequencer will trigger the transfer of video clips to Viz.

- **Find**, **Rename** and **Delete**.

> ⚠ **Note:** Viz Pilot 4 compatible playlists are not supported by Viz Pilot 5.4 and later.

DB Exported Tab

Lists all Media Sequencer playlists that have been saved to the database. Playlists are saved to the database by going to Playlist Menu **> Save to database**.

To **import** a playlist in Director that has been saved to the database, double-click a playlist. The playlist is opened and listed under the imported node by its save name.

> ⚠ **Note:** If the playlist has already been imported, a dialog appears that lets you overwrite the existing playlist or create a new copy.

Imported playlist are added to Media Sequencer where Director is connected.

8.16.2 Context Menu

Columns

- **Created:** Displays the Created column. This column shows the date and time that the playlist was created.
- **Modified:** Displays the Modified column. This column shows the date and time for the last changes done to the playlist.

- **Name:** Displays the Name column. The name column shows the name of the playlist. Name is enabled by default.
- **Profile:** Displays the Profile column. The column shows which profile each playlist is configured to use.
- **State:** Displays the State column. Possible states are Active or Inactive.
- **Auto Width:** When enabled (selected), Auto Width expands the columns and distributes them evenly.
- **Enable Sorting:** When enabled (selected), Enable Sorting enables the user to sort the list (ascending/descending).

Resources

- **New Playlist:** Creates a new playlist in the resource list.
- **New Folder:** Creates a new folder in the resource list.
- **Open:** Displays the selected playlist.
- **Open in New Window:** Displays the selected playlist in a separate window.
- **Activate in current profile:** Activates and monitors the selected playlist in the current profile, and enables the Media Sequencer to trigger transfer of video clips to Viz.
- **Deactivate:** Deactivates and stops monitoring of an activated playlist.
- **Find:** Displays the Resource Search.
- **Rename:** The selected playlist may be renamed.
- **Delete:** Deletes the selected playlist. Alternatively, use the keyboard key Delete (**Del**).

> ⚠️ **Note:** New Playlist, New Folder, Rename and Delete operations can only be performed on Director playlists or folders that group Director playlists.

8.16.3 Working with the Playlists Panel

Creating a New Playlist

1. Right-click the Playlists Panel, and from the Context Menu select **New Playlist**, or
2. Select **New Playlist** on the **File** menu.
   - This will set the focus on the new playlist.

Opening a Playlist

1. Double-click a playlist name in the resource list, or
2. Right-click the Playlist entry, and from the Context Menu select **Open**.

Adding a Data Element to a Playlist

1. Select the Data resource panel.
2. Drag and drop the data element(s) onto the playlist.
   - If the element is dropped over an existing element, the new element is positioned above the existing element.
   - If the element is dropped in an empty section of the Playlist, the new element is positioned at the bottom of the list.

3. Arrange the order of the elements using drag and drop.

Setting the Same Concept for the Whole Playlist

1. Open the Playlists.
2. Select the concept from the **Concept** dropdown menu on the toolbar.

> ⚠ **Note:** Data elements that do not support the selected concept will be colored yellow.

Selecting a Concept and Variant Using the Playlist

1. Open the Playlists where the Data element was added.
2. Select the Data element, and then select the new concept and variant from the element's **Concept** and **Variant** column.
   See also how To select a concept and variant.

8.17 Director Main Menu

8.17.1 File Menu

The File menu contains the following commands:

- **Reconnect to Database:** If the connection to the database is lost, click **Reconnect to Database** to reestablish the connection. It is also recommended to reconnect if the application has been running unattended for a long time.
- **Refresh Scripts:** Reloads all the scripts in the database.
- **New Playlist:** Creates a new Playlist Window.
- **View log:** Opens the View Log dialog box.
- **Exit:** Shuts down Director.

8.17.2 Edit Menu

The Edit menu contains the **Cut**, **Copy** and **Paste** commands, which can be used to edit text.

8.17.3 Options Menu

The Options menu contains the following commands:

- Data List: Displays the data elements in the database:
  - **Show Last Two Weeks:** Displays data elements created the last two weeks.
  - **Show All:** Displays all data elements.
  - **Show X Day(s):** Displays data elements that have been used during the previous X number of days.

    ⚠ **Note:** X number of days is set under the **General** tab in the Director Preferences window.

- **Delete Data Items:** Opens the Delete Data Items dialog box.
- **Keyboard Configuration:** Opens the Keyboard Configuration dialog box.
- **Preferences:** Opens the Director Preferences dialog box.
- **Hints:** Enables/disables hints in the Status Bars. Hints are displayed when the pointer is hovered over items such as the Viz Gateway connection icon. Hints are enabled by default.
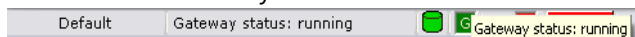
Delete Data Items
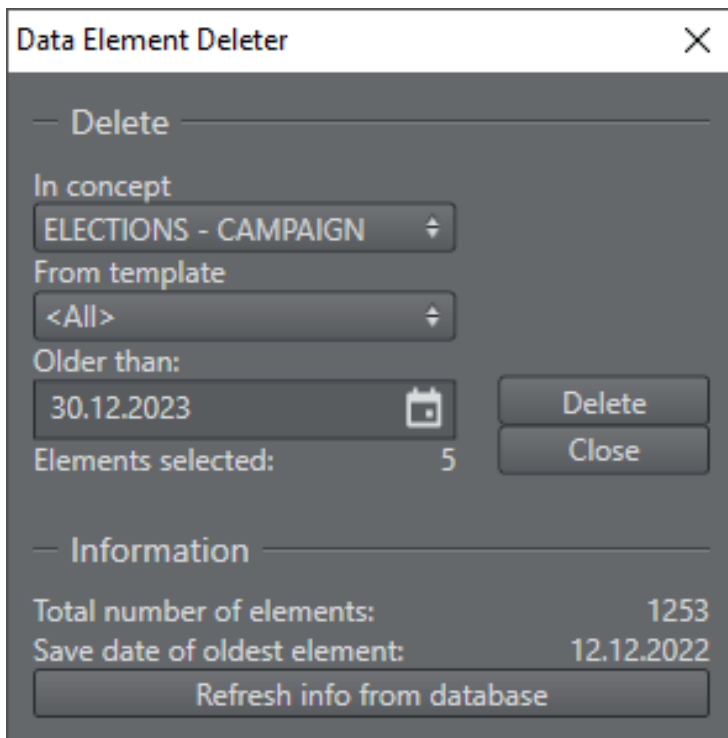
This function permanently removes data elements from the database that are not tagged as Library elements. Library elements must be deleted manually.

- **In concept:** Sets the concept from where the data elements should be deleted. Selecting **All** deletes all data elements of all concepts for all templates that are not saved as library elements.
- **From template:** Sets the template from where the data elements should be deleted. Selecting **All** deletes all data elements of templates that are not saved as library elements within the selected concept.
- **Older than:** Deletes data elements that are older than the given date apart from those tagged as Library elements. This date is by default set one week back in time.

The **Information** section updates on opening the dialog box, and when clicking **Refresh info from database**.

8.17.4 Tools Menu

The Tools menu contains the following commands:

- **Camera Control (CTRL + W):** Opens the Camera Control dialog box.
- **Import MOS message file...:** Opens the *Select MOS Message File To Import* dialog box. This enables the playlist to be sent to the Media Sequencer through Director when it is necessary to manually export a playlist from a newsroom system.
  Browse for and select the exported newsroom playlist to be added to the Media Sequencer, and click **Open**.
- **Template Wizard:** Opens Template Wizard. This menu option is only enabled when Template Wizard (template design tool) is installed alongside Director.
- **Show Command Console:** Opens the Commands window, see Show Command Console.

- **Media Sequencer Configuration:** Opens the Media Sequencer Configuration window. All configuration options available through this window are stored in the Media Sequencer.
- **Profile Configuration... (CTRL + M):** Opens the Profile Configuration window. Profile Configuration is used to configure the Viz profiles that are used for playing out graphics and video.
- **Post Render Device:** Opens the Post Render Device window.
- **TimeCode (TC) Monitor:** Opens the Timecode Monitor window. This option is only available if video playout is integrated.
- **Export:** Opens the Export from Director window.
- **Import:** Opens the Import into Director window.

Camera Control

Virtual cameras are set up in Viz Artist. The Camera Control in Director controls the virtual cameras, so that image sequences may be used together with the templates. In a virtual studio system, this window will call matrix setups as they are defined in the Viz IO setup. If no virtual studio is available, button 1: 16 will cut between virtual cameras 1: 16.

Select **Camera Control** on the **Tools** menu, or press the **CTRL + W** keys to use the Camera Control.

8.17.5   Viz Engine Menu

The Viz Engine menu contains the following commands:

- **Show CMD ON (SHIFT + CTRL + C):** Shows the Viz Engine console window on the program channel, and all verbose commands being sent and received.
- **Show CMD OFF (SHIFT + CTRL + F):** Hides the Viz Engine console window on the program channel.

- **Cleanup Renderers:** Unloads all scenes, images, and fonts from both program and preview channels, and resets them to their initial state.
- **Commands:** Controls the playout of the loaded scene:
    - **Start:** Runs the scene on the selected program channel.
    - **Continue:** If the scene contains stop points, and the current state is at a stop point, click Continue to run the animation from this stop point until the next.
    - **Stop:** Stops the running animation.
    - **Reset:** Resets the animation to its start state.
- **Clear:** Clears the layers of the loaded scene:
    - **All Layers:** Clears all layers.
    - **Front Layer:** Clears the front layer.
    - **Middle Layer:** Clears the middle layer.
    - **Back Layer:** Clears the back layer.
- **Local Viz Engine Preview:** Opens a Local Preview in Director window.
- **Remote Preview (CTRL + P):** Opens a Remote Preview in Director window.

8.17.6   Playlist Menu

This menu option is only visible when a playlist is open. In Director it is possible to import a playlist from an XML file. The XML file must be created with references to templates and data that will be inserted into the playlist, thus creating new data elements during import. A template reference is a reference to the external ID registered for a template using Template Wizard's template manager tool.

The Playlist menu contains the following commands:

- **XML Filler:** Use this option to import a playlist file (XML file).
    - **Open:** Creates a new playlist based on the content of the XML file.
    - **Add:** Appends the content of the file to a new or existing playlist.
- **Clear Playlist:** Clears the playlist by removing all elements in it.
- **Save to database:** Saves any playlist held by the Media Sequencer to the database. Imported playlists are added to the Media Sequencer that Director is currently connected to.
  To **import** a playlist that has been saved to the database, go to the Resource Panel > Playlists tab > DB Exported Tab. Double-click the **exported playlist**. The playlist is opened and listed under the imported node by its save name.

XML Example

The XML example below contains the name of the playlist, a group and three elements with an external ID (`templateID`). External IDs are configured in Template Wizard's template manager tool per template.

```xml
<?xml version="1.0"?>
<playlist>
    <name>Name of playlist</name>
    <group visible="true" name="Group name">
        <event>
            <templateID>100</templateID>
            <dataName>Data Element Name 001</dataName>
```

```
            <comment/>
            <data>
                <field>FirstName LastName 001</field>
                <field>Designation 001</field>
            </data>
        </event>
        <event>
            <templateID>100</templateID>
            <dataName>Data Element Name 002</dataName>
            <comment/>
            <data>
                <field>FirstName LastName 002</field>
                <field>Designation 002</field>
            </data>
        </event>
        <event>
            <templateID>100</templateID>
            <dataName>Data Element Name 003</dataName>
            <comment/>
            <data>
                <field>FirstName LastName 003</field>
                <field>Designation 001</field>
            </data>
        </event>
    </group>
</playlist>
```

8.17.7   Windows Menu

The Windows menu options are commonly used to quickly cascade or tile the playlist windows.

The Windows menu contains the following commands:

- **Cascade:** Arranges all open playlists into a cascaded stack, and displays the playlists in the Work area.
- **Tile:** Arranges all open playlists horizontally from the top of the screen to the bottom, and displays an equal part of all the playlists.
- **Hide resource bar (CTRL + B):** Hides the Resource Panel. Alternatively, use the keyboard shortcut **CTRL + B**. The Control panel is displayed by default.

> ⚠️ **Note:** The Windows options affect other windows as well.

8.17.8   Help Menu

The Help menu contains the following commands:

- **Director Help (F1):** Opens the online version of the Viz Pilot User Guide (this document). A local copy can be accessed from the Windows Start Menu.
- **About:** Opens the About window that contains information about the installed Viz Pilot version. A link to the list of Third Party Component Credits is also available.

8.18   Export And Import In Director

This section contains the following topics:

- Exporting from Director
- Importing into Director

8.18.1   Exporting from Director

In the Pilot Exporter window, playlists, templates and data elements can be exported to file. This option supports export of all the included templates, concepts, scripts, standalone and transition logic scenes. Data elements can be exported as part of a playlist, or individually. Playlists and all parts of the templates are exported to an XML file, and the related scenes are exported as a Viz archive file (.via).

This section contains the following topics:

- Pilot Exporter Window
- Example Export Folders

Pilot Exporter Window

> ⚠️ **Note:** Pilot Exporter is not available while using the Legacy Playlist. To enable Pilot Exporter, you must first enable the new Playlist Window. Go to **Options > Preferences > Playlist > General > Use playlist from <= 5.7** and uncheck this option (see Playlist - General Preferences).

To Export

1. Configure a Viz preview channel in Profile Configuration.
2. Select **Tools** > **Export** to open the **Pilot Exporter** window.
3. Drag Playlists and Templates from the Resource Panel into the **Pilot Exporter** window, and click **Next**.
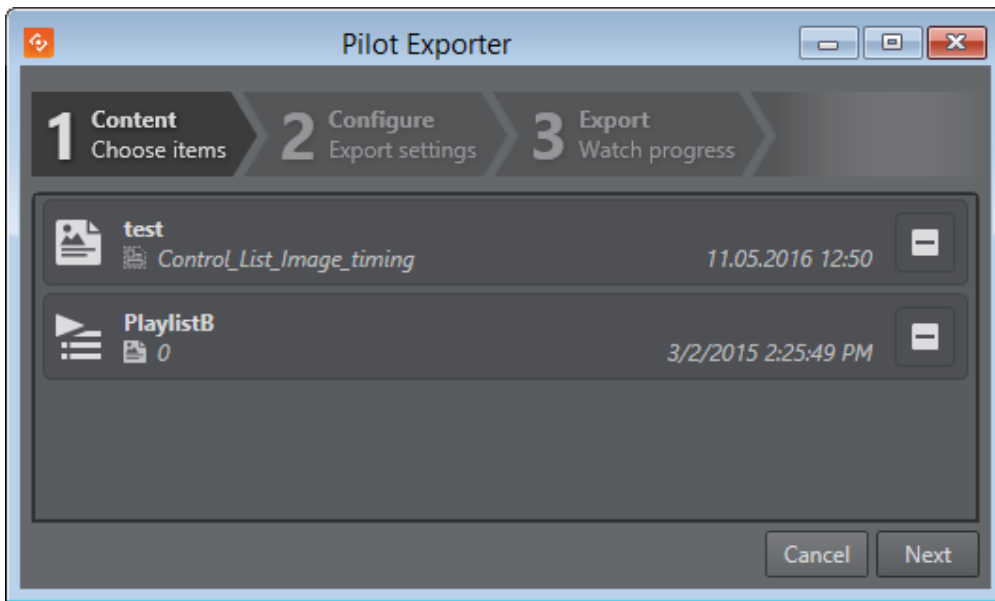
4. Enter a path and filename for the exported XML file (see Example Export Folders).
   Optional: Check the **Create Viz Archive Export** option to export Viz scenes. Enter a path and filename for the exported Viz archive.

> ❌ **IMPORTANT!** The Viz Engine archive file can only be created if the Viz Engine is connected and On Air, and the specified path exists on that machine.
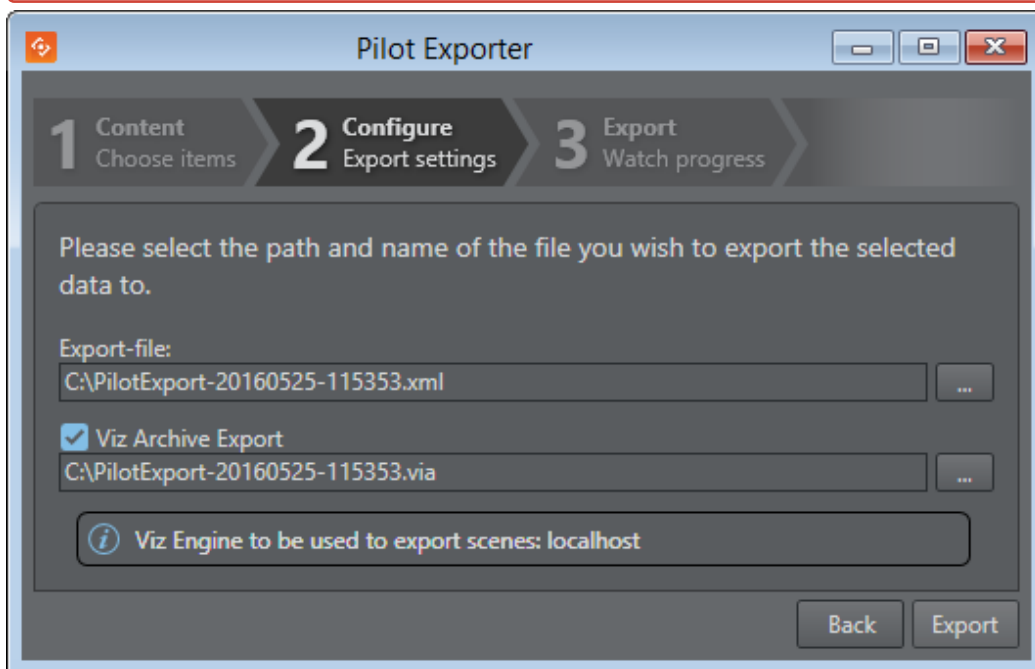
Viz Engine to be used to export scenes: The Viz Engine machine on which the Viz Archive Export will be created. Exporter selects the Viz Engine according to the following rules:

- If the currently active profile has a preview channel defined, use that Viz Engine, otherwise use the program channel.

- If the channel is configured with multiple Viz Engines, use the one at the top of the list.
- If the profile is modified after the Export window has been opened, but before **Export** is clicked, then the export runs against the reconfigured channels.

> ⚠️ **Note:** Changing the profile settings during the actual export will not change which Viz Engine the export uses.

5. Click **Export**, and then check the progress and status of your export.
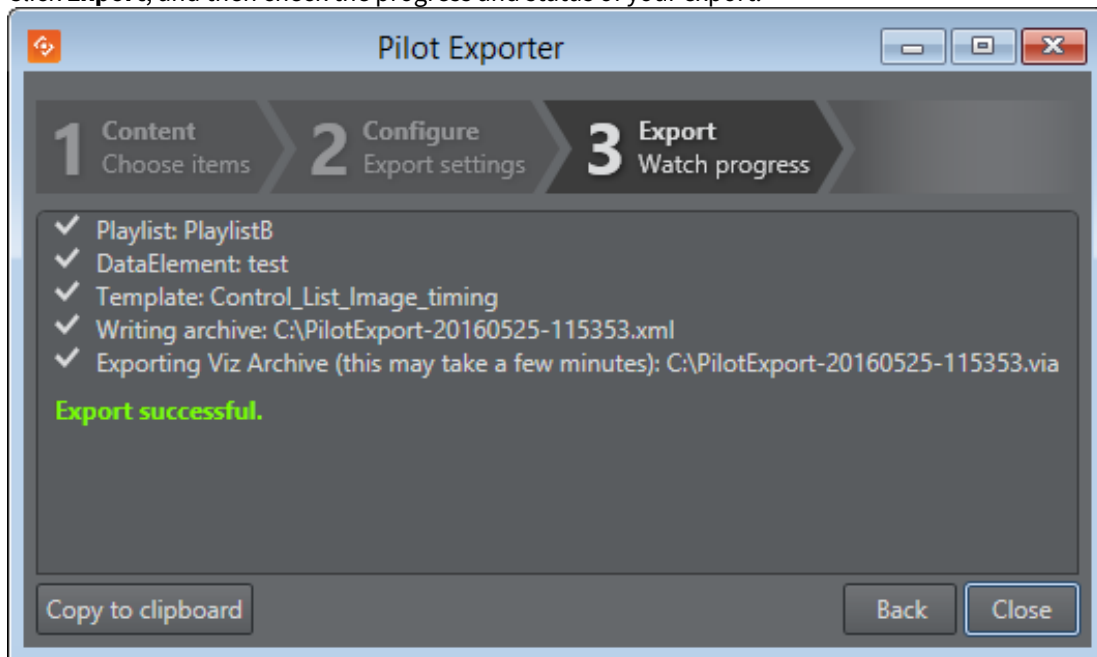
6. Click **Close**.

Example Export Folders

Export of templates and scenes will be exported to different folders, unless a **shared export folder** is mapped on both the control and render machines.

Normally, the Pilot archive file (.xml) will be placed on the user's machine, and the Viz archive file (.via) will be placed on the Viz Engine (see the table below).

When exporting graphics, a preview channel must be configured and used.

Path	Filename	Comments
C:\temp\	archive.xml	**Pilot archive filename** . In this example, the temp folder must be present on the users machine, and the user must have write access.

Path	Filename	Comments
C:\temp\	archive.via (archive) (archive.eva)	**Viz Engine archive filename** .The Viz Engine archive path automatically defaults to match the Pilot export path, above. In this example, the temp folder must be present on the Viz Engine machine, and the Viz user must have write access. If the file extension is missing Viz will automatically add one (for example archive.via). If the file extension is not correct according to Viz, Viz will replace the extension with the correct one (for example archive.txt will become archive.via). Note that there is no feedback from the Viz Engine if the path does not exist and the file is not created.
Not specified	archive.via	In this example, the archive will be placed on the Viz Engine, in the Viz 3.x program folder: Windows 7: `%LOCALAPPDATA%\VirtualStore\Program Files\Vizrt\Viz3}}`Windows XP: `{{%ProgramFiles%\Vizrt\Viz3`
\\host\	archive.via	The Viz machine must have write access to the UNC path.

8.18.2   Importing into Director

In the Pilot Importer window, playlists and data elements can be imported from file, including the templates, concepts, scripts and transition logic scenes that are required by them.

> ⚠ **Note:** The folder structure used for organizing the templates must be the same in the import environment as it was in the export environment. In addition, resources (such as images used by data elements etc.) must be accessible in the same location (file path) in both export and import environments.

Pilot Importer Window

Importing Playlists, Templates and Scenes

1. Select **Tools** > **Import** to open the **Pilot Importer** window.
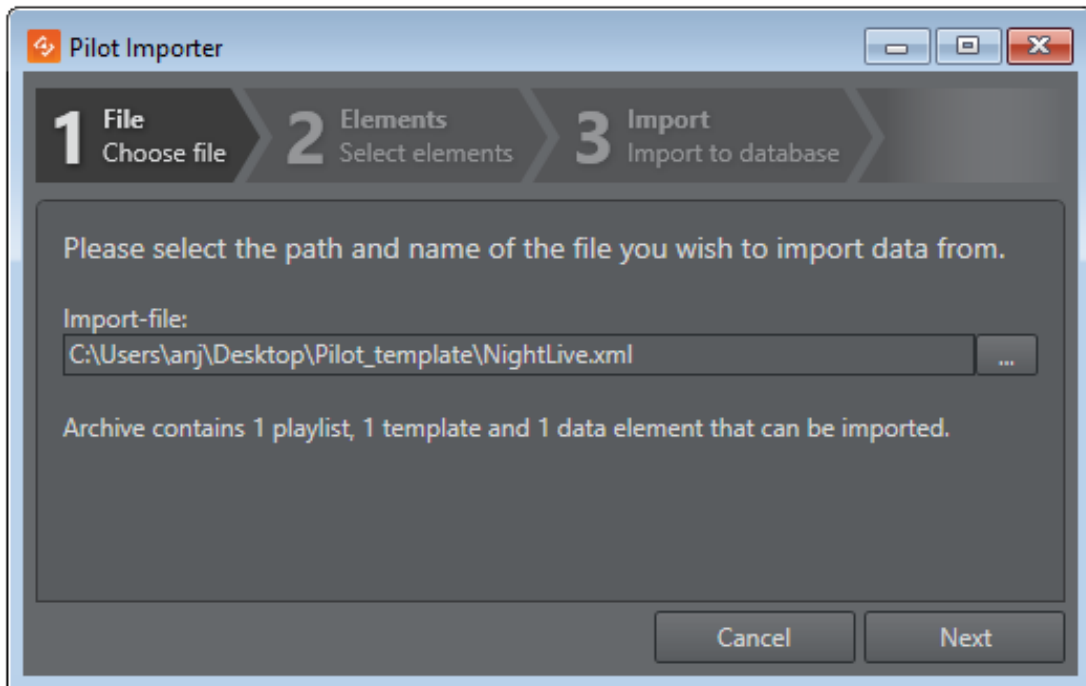
2. Browse for and select the Pilot archive file (.xml).

⚠️ **Note:** Scenes cannot be imported using the Pilot Importer when they are in Viz archive (.via) file format. However, scenes can be imported directly into Viz Artist or Graphic Hub. See the Viz Artist User Guide and Graphic Hub User Guide.

3. Click **Next**.
4. Select the templates, data elements and playlists to be imported. In the search toolbar, the user can filter on templates, data elements and/or playlists using the designated buttons, or type in a search based on name, modified date or template name.
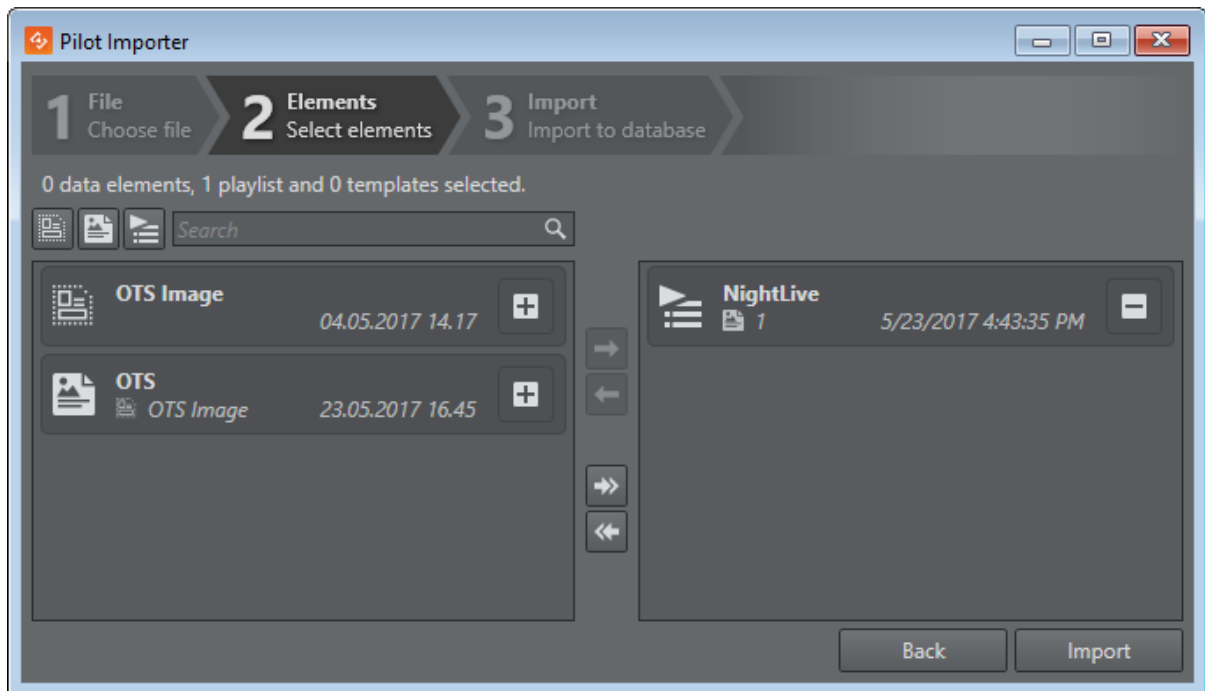
5. Click **Import**, and then check the progress and status of your import.
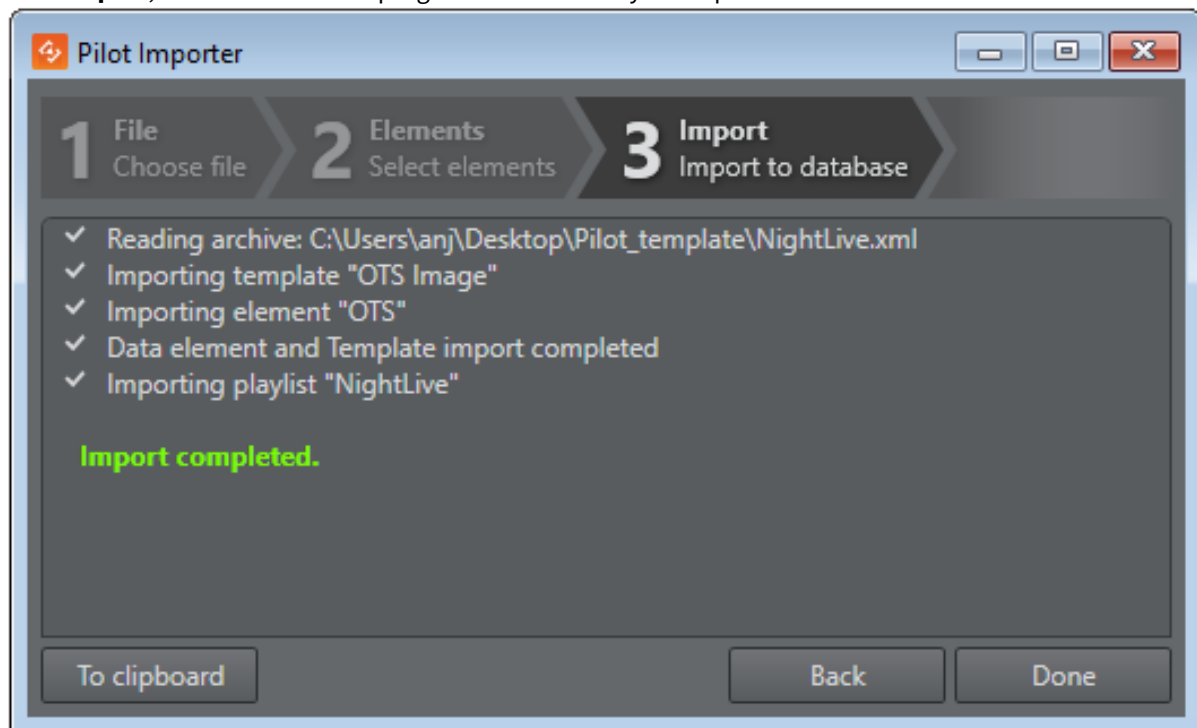
6. Click **Done**.

> **ℹ Info:** A confirmation dialog appears if the user tries to cancel or close an ongoing import or export job.

9 Template Wizard

Template Wizard is a tool to auto-generate basic templates, and creating customized graphics templates using a rich set of components combined with advanced client-side scripting logic.

Templates for Viz Pilot are created in Template Wizard as custom control and content entry interfaces for graphics scenes. The templates can be customized with a range of components and scripting tools.

This section contains the following topics and procedures:

- Workflows
- Viz Engine Preview in Template Wizard
- Categories and Channels
- Keyboard Configuration and Macros
- Component Palette
- Export and Import in Template Wizard
- Toolbar
- VCP Resource Panel
- Viz Pilot News Test Page
- Update Script Editor
- Execution Logic Editor
- Getting Started
- Main Menu
- Script Editor
- Wizard
- Template Editor

9.1 Workflows

9.1.1 Broadcast

The basic workflow from graphics design to a working template:

- Scenes are made in Viz Artist. Multiple scenes can make up one template, categorized under different Variants.
- Objects in the scene that the graphics designer wants to make editable in the finished template are given the appropriate control plugins.
- Templates are made in Template Wizard, after importing the scenes over.
- Exposed objects from the scene are used to automatically create a basic template.
- The template designer creates additional logic using graphical components, and two types of script logic:
    a. The Script Editor makes it possible to add scripts inside the template to aid users in filling out or using the template in a best-practice way.
    b. The Update Script Editor allows the designer to add scripts that will be executed by Update Services on the server side, and not on the client side through Director.
- The template is tested and saved, preferably to a descriptive concept, and assigned to a category using the Template Manager tool. (Concepts group templates that are used together in a show or broadcast).
- Templates are made available to the newsroom and control room systems.
- Using Viz Pilot News, a journalist or editor fills the graphics template with any information required to make up a complete graphic. This can include: text, numbers, images, person information, videos (from the Viz One system) and maps (from the Viz World system). The template is saved into the Viz Pilot system creating a data element that is added to the newsroom story.
- The Viz Pilot operator monitors the newsroom playlist (rundown) and plays the graphics on-air at the correct time in the broadcast. This can be achieved by external GPI commands, keyboard shortcuts, button boxes, cursor clicks and so on.

9.2 Viz Engine Preview In Template Wizard

This section contains the following topics:

- Remote Preview in Template Wizard
- Local Preview in Template Wizard

9.2.1 Remote Preview in Template Wizard

Using a Viz Engine installed on another remote machine is the recommended preview mode for Template Wizard.
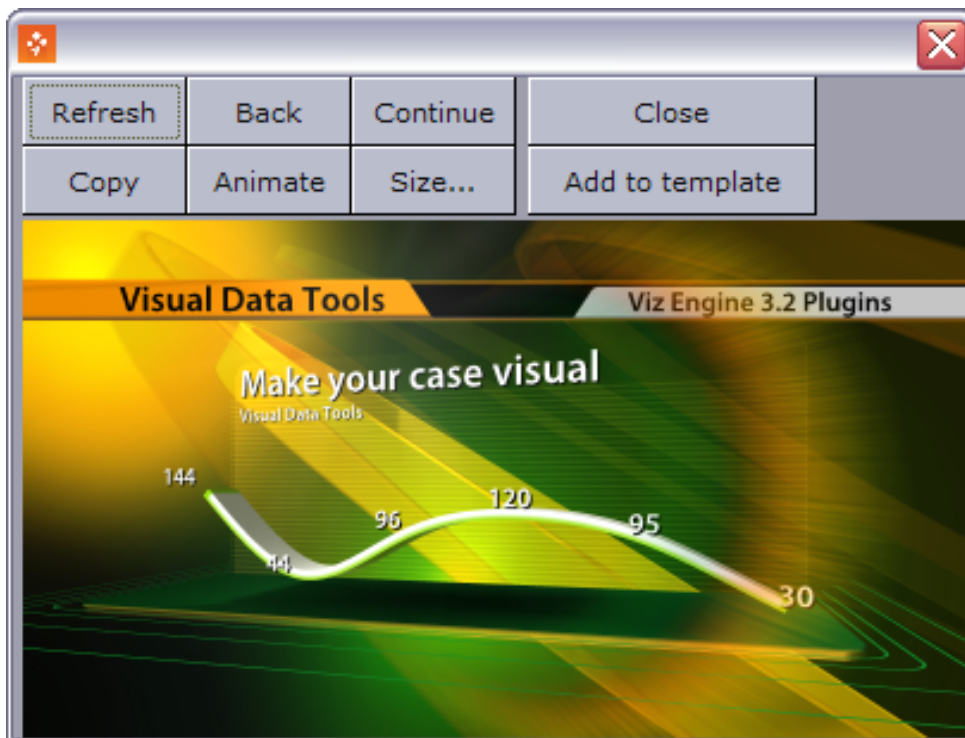
Remote Preview is the default mode for Template Wizard. Unlike Local Preview, remote preview does not require a local installation of Viz Engine and is therefore the recommended option. For more information on configuring Remote Preview, see Configuring Remote Preview in Director.

> ⚠️ **Note:** The remote preview of templates with video clips requires a full Viz One video workflow and the use of Viz Pilot in order to trigger the transfer of video clips to Viz Engine.

Properties and Parameters

- **Refresh:** Refreshes the scene preview displaying a snapshot of the scene at its current animation position. This enables you to fetch new snapshots of the scene while it is being rendered on a remote Viz Engine without seeing the full animation in the preview window.
- **Back:** Sets the animation back to the last stop point.

- **Continue:** Continues the animation to the next stop point.
- **Copy:** Copies the snapshot to the clipboard.
- **Animate:** Animates the scene in the remote preview window. The animation is based on a series of snapshots of the scene rendered on the remote Viz Engine. Click **Animate** to start and stop the animation.
- **Size:** Sets the size of the preview window. There are two default settings, one for PAL and one for NTSC, and one custom.
- **Close:** Closes the preview window.
- **Add to template:** Adds a snapshot of the previewed graphics to the template. The snapshot is appended to a Static Image Component component.

Previewing a Template

1. Open a template and change to **Run mode**.
2. Select **Preview** on the **View** menu, or press **CTRL + P** to open the preview window.
3. Click the **Load (F7)** button and then the **Refresh** button in the Viz Engine window to see a snapshot of the rendered scene.
4. Click **Animate** to see a series of snapshots.

9.2.2   Local Preview in Template Wizard

Local Preview must be selected by enabling the Viz Engine Preview plugin in the Changing Settings. For more information, see Configuring Local Preview. The interface is the same as that used in Viz Pilot News and Director, see Local Preview in the Newsroom.

> ⚠️ **Note:** This mode requires that a Viz Engine is installed locally on the Template Wizard client machine for preview purposes and is therefore not the recommended preview option.

> ⚠️ **Note:** The remote preview of templates with video clips requires a full Viz One video workflow and the use of Viz Pilot in order to trigger the transfer of video clips to Viz Engine.

Previewing a Template in Local Preview Mode

1. Open a template and change to **Run mode**.
2. From the main menu select **Viz Engine** > **Local Preview**.
3. In Template Wizard click the **Load** button to load and preview the scene. The **Start/Take** button in the Local Preview window does not load the scene on Viz Engine. Once loaded by Template Wizard, the scene can be previewed using the control buttons in the Local Preview window.

9.3   Categories And Channels

The Categories and Channels option on the Tools menu opens an editor window for adding template category and channel assignments for those categories.

9.3.1   Categories and Channels Editor
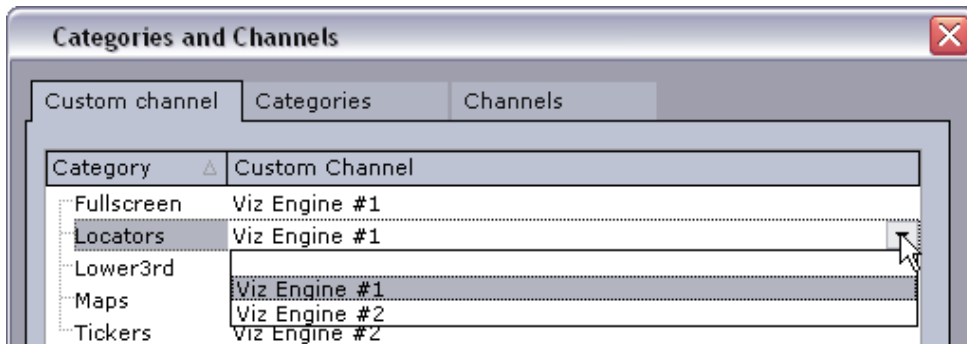
The channel name must correspond to the output channel configured in Director. In some situations a user might want certain types of graphics to be played out on a specific output channel, for instance lower thirds on one channel and full screens on another. By adding a template to a specific category, the category's channel is set as the default output channel when the template's data elements are added to a playlist in Director.

> ❌ **IMPORTANT!** Legacy templates that do not use Control Object are unable to assign themselves to the correct program channel. These templates will therefore always default to the default program channel in the profile configuration, and disregard information in the Categories and Channels setup.
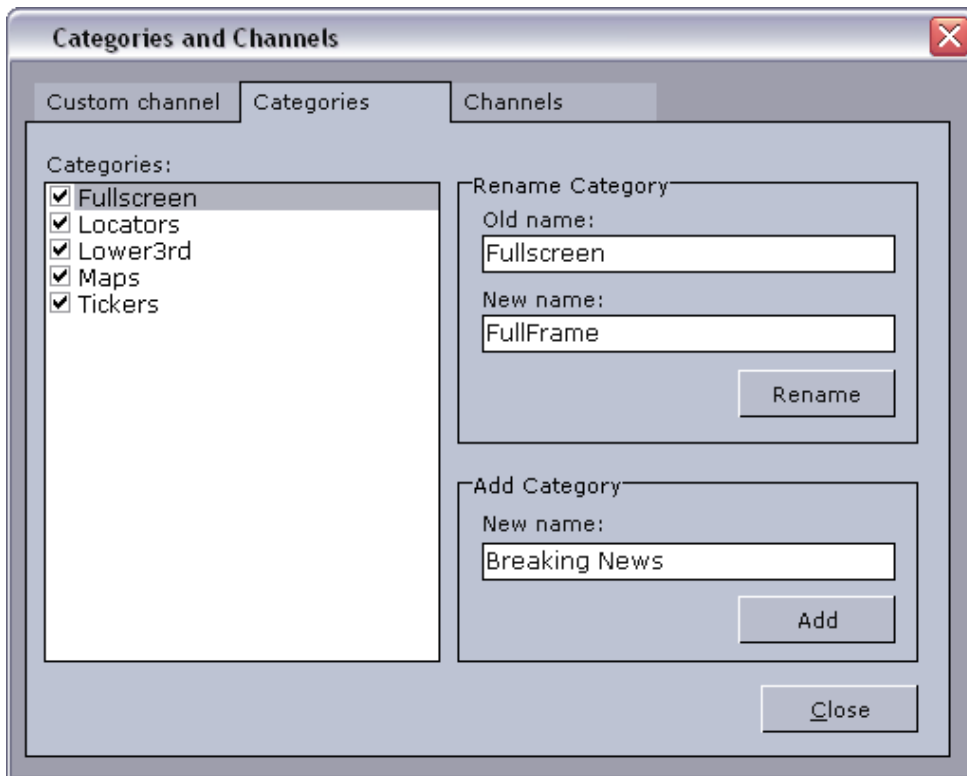
- **Custom channel:** Sets the default channel for each of the defined categories.
- **Categories:** Adds and renames categories.
- **Channels:** Adds and renames output channels. Note that the channel name **must** correspond with the channels configured in Director.

For more information on how to set categories for templates, see the Template Manager's Variants editor.

> ℹ **Info:** To pass a default channel to a newsroom system, the parameter `ax_add_itemchannel` in Pilot Data Server's VCP Parameters must be enabled.

9.3.2 To Delete a Category or Channel

Select a category/channel and press **Delete**.

9.4 Keyboard Configuration And Macros

This section contains the following topics:

- Keyboard Configuration
- Template Wizard Macro Commands

9.4.1 Keyboard Configuration
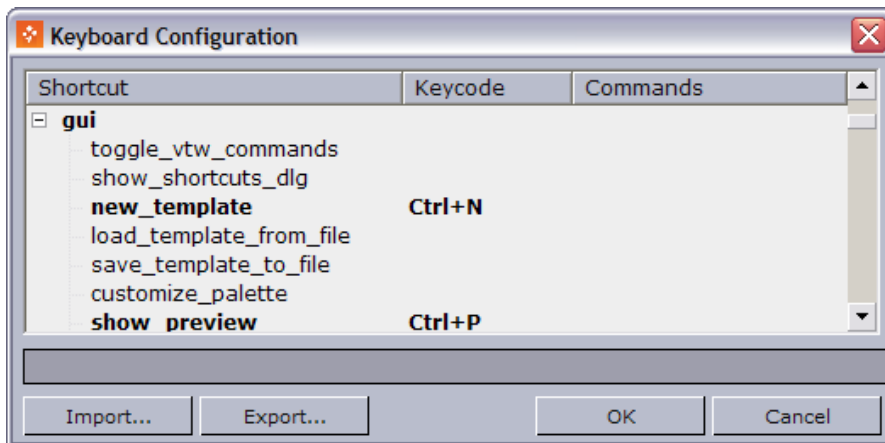
Select and define macros and assign them to specific key combinations, predefined functions and VB Scripts. To open the **Keyboard Configuration** window, select **Customize Keyboard** on the **Tools** menu.

This section contains the following topics:

- Keyboard Configuration Window
- Working with macros and shortcuts
- Macro Shortcut
- Predefined Functions for Keyboard Configuration

Keyboard Configuration Window

Columns

- **Shortcut:** Displays the macro name.
- **Keycode:** Displays the keyboard shortcut.
- **Commands:** Displays the shortcut command.

Buttons

- **Import:** Opens the **Open** window where a previously exported KBD (keyboard) file can be imported.
- **Export:** Opens the **Save As** window where the current settings can be exported as a KBD file.
- **OK:** Saves the keyboard shortcut configurations, and closes the window.

  File location: `%PROGRAMDATA%\Vizrt\Viz Pilot x.y\Config\vtw_keyboard.kbd`

- **Cancel:** Closes the window.

Context menu

- **Add Macro:** Opens the Macro Shortcut window, where a new macro can be created.
- **Delete Macro:** Deletes the currently selected macro. Alternatively, click the keyboard shortcut **CTRL + DELETE**.
- **Remove Shortcut:** Deletes the shortcut key code of the currently selected macro. Alternatively, use keyboard shortcut Delete.

Working with Macros and Shortcuts

- **Add a macro command:** Select **Add Macro...** .from the context menu to open the Macro Shortcut window, then enter the key combination, macro category:macro name and commands.
- **Edit a macro command:** Double-click it to open the Macro Shortcut window.
- **Delete a macro command:** Select it and then either press the **CTRL + DELETE** keys, or select **Delete Macro** from the context menu.
- **Remove a macro command shortcut:** Select the macro and then either press the **DELETE** key, or select **Remove Shortcut** from the context menu.

Macro Shortcut

The New/Edit Macro Shortcut window is opened by selecting **Add Macro** from the Keyboard Configuration Window context menu, or double-clicking an existing macro.

The Macro Shortcut window contains the following fields and buttons:

- **Enter Key:** Sets the keyboard shortcut for the macro. Place the cursor in this field and press the desired shortcut keys to set the key value.
- **Macro Name:** Displays the macro category (for example macros:, gui:, viz:) and the name of the macro. If a folder was selected when the Macro was opened, the category name will automatically be displayed in the

Name text box. The category name can be set or changed in the Name field by adding the category and macro name separated by a colon (<macro category>:<macro name>).

> ⚠ **Note:** If a category is not added to the name it will be placed in the macros category. It is also possible to add new categories by defining them in the macro (for example foo:bar).

- **Global:** Makes the macro work including when a template is open.
- **Description:** Description of the macro.
- **Macro Commands:** The names of the selected Predefined Functions for Keyboard Configuration are displayed.
- **VB Script:** A text field where script code can be entered. This provides the possibility to further enhance and adapt the commands.

> ❌ **IMPORTANT!** The function stub named *RunScript* must be used as the main script or entry point for other scripts; hence, it cannot be renamed or removed.

- **Show Macros:** Opens the Predefined Functions for Keyboard Configuration window.
- **Save As New:** Saves the macro as a new instance.
- **OK:** Saves the macro, either as a new or modified macro.
- **Cancel:** Closes the window, without saving any changes.

Predefined Functions for Keyboard Configuration
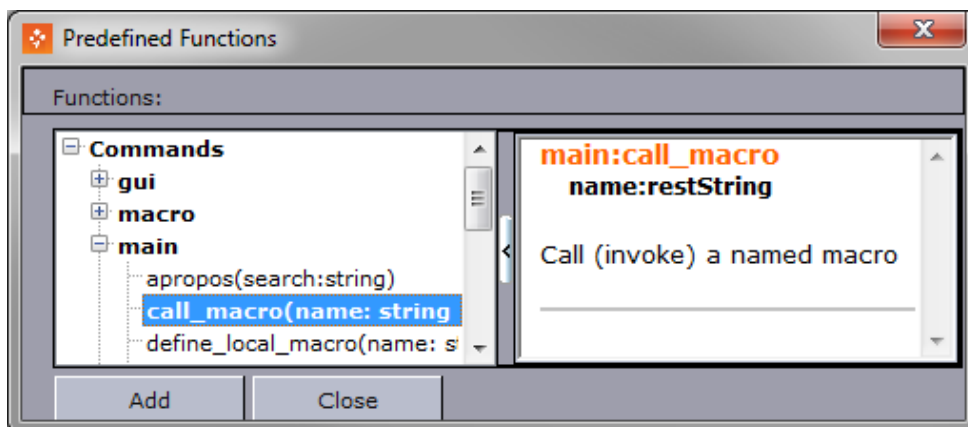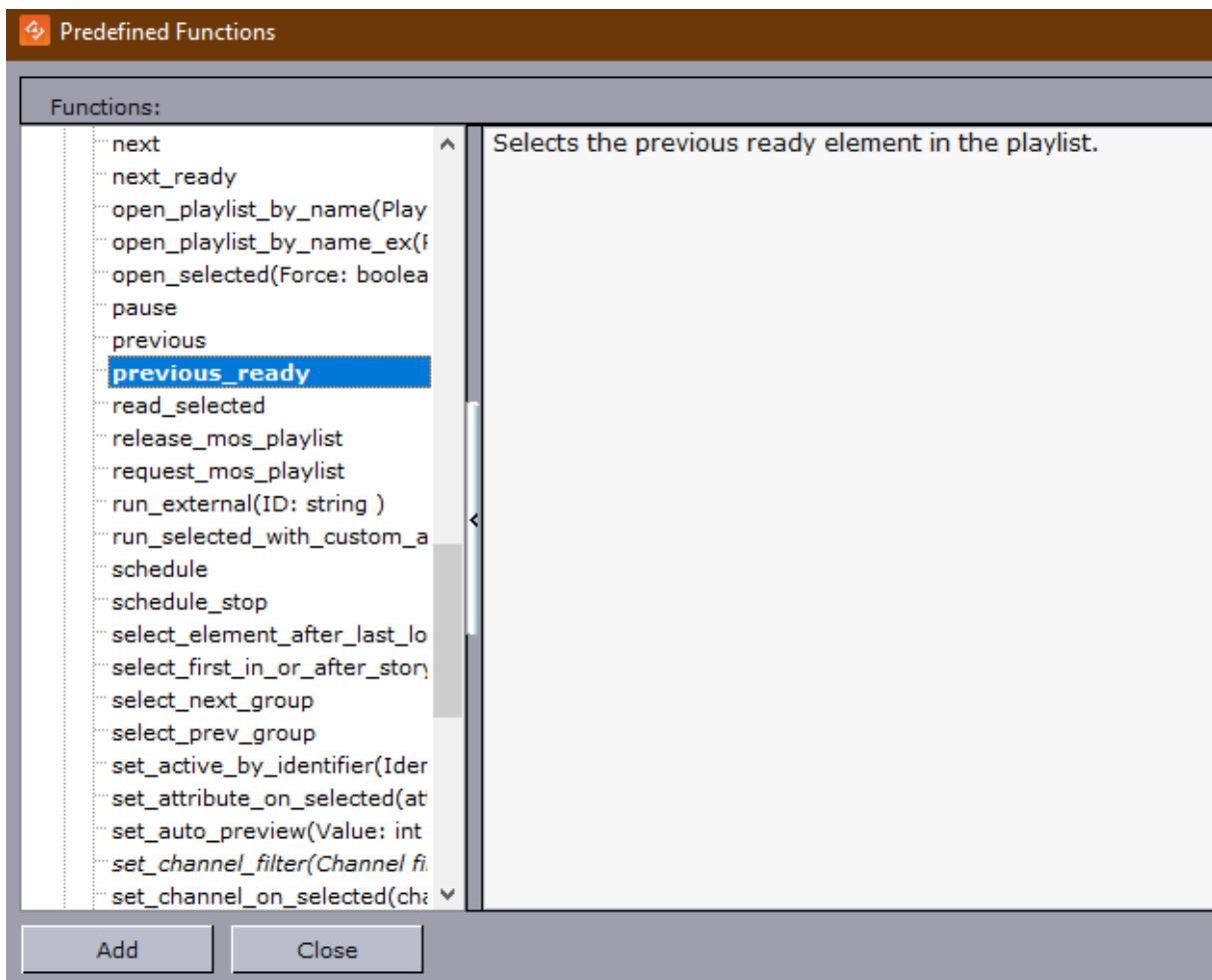
The **Predefined Functions** window contains a list of commands that can be added to a keyboard shortcut.

> ⚠️ **Note:** In the screenshot above, **ready** refers to the status of MOS ObjAir status (see below).

next_ready

Selects the next element in the playlist with the attribute MOS ObjAir set to "READY".

previous_ready

Selects the previous element in the playlist with the attribute MOS ObjAir set to "READY".

> ⚠️ **Note:** Only commands relevant to creating shortcuts are presented in this window. A more extensive list of predefined functions is available when using the Script Editor (see Script Editor Toolbar).

9.4.2   Template Wizard Macro Commands

Execute predefined functions without using keyboard shortcuts and macros.
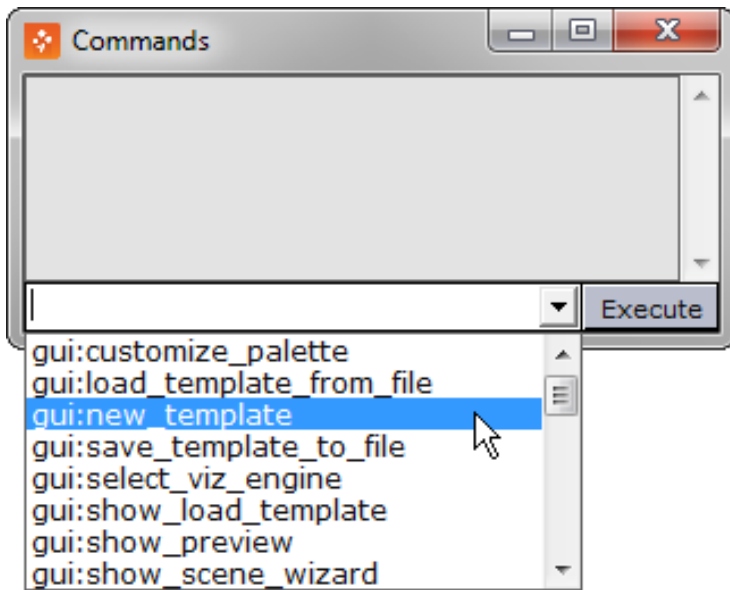
9.5   Component Palette

This section contains the following topics:

- Selecting Components
- Adding a Component
- Changing a Component

9.5.1   Selecting Components

A standardized subset of components is available from the dropdown list. The full list of components can be viewed by selecting **Tools > Component Palette**.
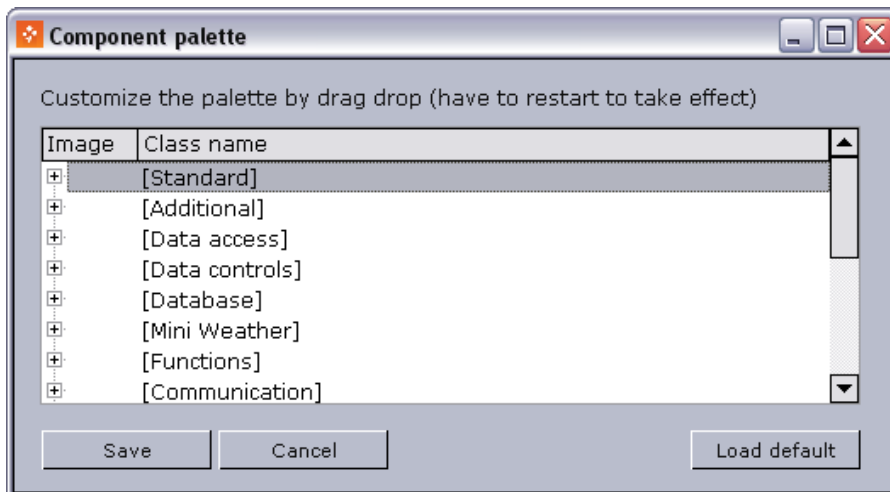
- The Component Palette lists all of the available components. Organize your own palette by dragging items around. Components added to the Unused category will not be visible in the toolbar.
- Any customizations to the component palette are saved to: `%PROGRAMDATA%\Vizrt\Viz Pilot x.y\Config\custom_components.cfg.`
- To return to the default settings, click **Load default**.

9.5.2   Adding a Component

Components that are used in conjunction with scripting, such as database components, dialog boxes, com ports, etc., are added by selecting them from the components toolbar and clicking once on the template.

1. Select a component palette (for example **Standard**) from the dropdown list on the menu bar.
2. Click a component button.
3. Click once on the template to add it to the template.

9.5.3 Changing a Component
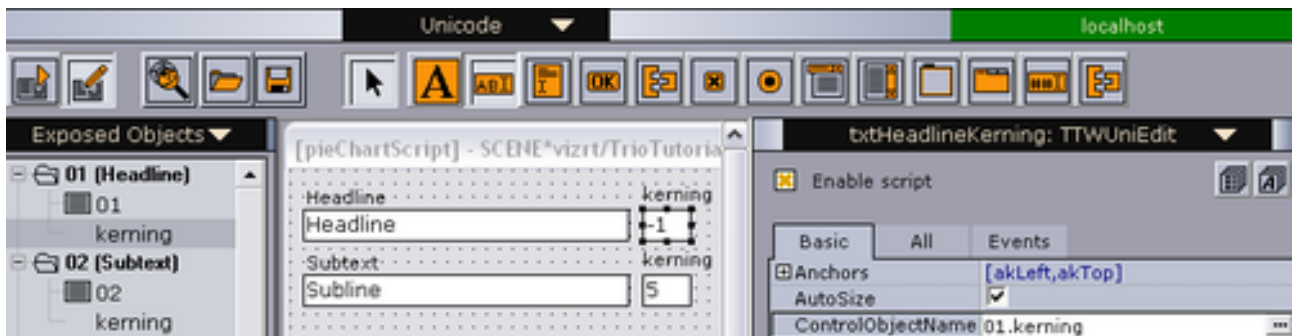
1. Delete the existing component on the template.
2. Select a new component from the component palette, and click once on the template to add it to the template.
3. From the Exposed Objects pane, select and drag and drop the object's exposed property onto the newly added component.

9.6   Export And Import In Template Wizard

Export and import in Template Wizard is similar to export/import in Director, with some differences related to playlists:

To import playlists, you must first specify a Media Sequencer for Template Wizard to connect to using the ini file or the command line. If no Media Sequencer is specified, Template Wizard will attempt to connect to the localhost for the playlist import. See Specifying Command Line Parameters for Template Wizard.

> ⚠️ **Note:** In order for the Export tool to work correctly, the Viz Engine must be On Air *before* Template Wizard starts.

For a full description of the import and export tools, see Export and Import in Director.

9.7   Toolbar

A Single toolbar operates in two different modes, one for template design (Design mode) and one for testing (Run mode).

In addition, the Viz Indicator on the toolbar shows Template Wizard's connection status to Viz Engine.

The **Run/Design button** toggles the two modes:

- **Run (F9)**: Switches to Run Mode. This mode requires that a template is open.
- **Design (F9)**: Switches to Design Mode. This mode is the default mode.

⚠️ **Note:** Toggle the run and design modes by using the F9 key.

9.7.1   Run Mode

- **Load**: Loads the scene on Viz and starts the preview.
- **Continue**: Continues the scene animation if the scene contains stop points.
- **Take Out**: Takes the scene off-air.
- **Back**: Resets the scene to its starting position.
- **Refresh**: Refreshes the scene with new data without reanimating the scene.
- **Preview**: Loads the scene on Viz and starts the preview.
- **Camera (1-4)**: Previews the scene from 4 camera angles (configured in Viz).

⚠️ **Note:** Refresh and Preview requires scenes that use Control plugins.

9.7.2   Design Mode

- **New**: Opens the Wizard to select scenes for automatic template creation.
- **Open**: Browses for and opens an existing template.
- **Save**: Opens the Save Template dialog.

Viz Indicator

The Viz indicator in the upper right corner shows which output machine Template Wizard is connected to.

9.8   VCP Resource Panel

The resource panel lists all available templates, and the exposed objects or scene objects when a template is opened.

This section contains the following topics:

- Templates
- Exposed Objects
- Scene Objects
- Layer Panel

9.8.1   Templates
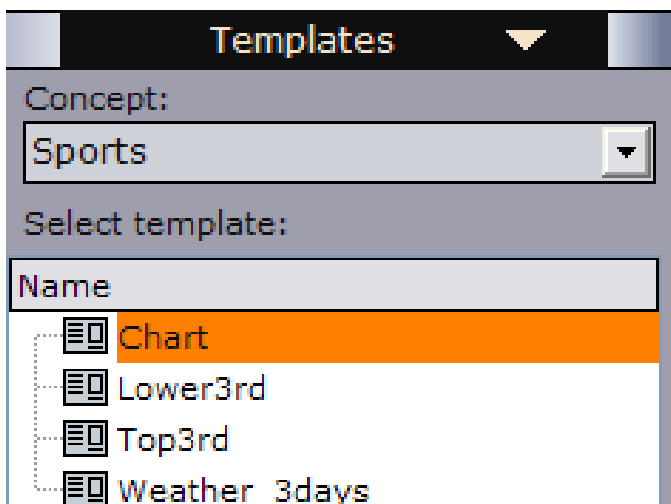
- **Concept:** Lists all concepts. Available options are All, Unassigned and all visible user created concepts. Hidden concepts must be made visible by checking them off in the Template Manager.
- **Name:** Lists all templates for the selected concept.

Templates Context Menu
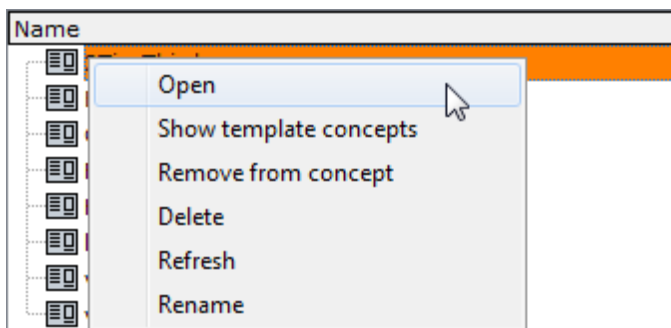
- **Open:** Opens the template in the editing area.

- **Show templates concepts:** Opens the template manager tool displaying the selected template's associated concepts.
- **Remove from concept:** Removes the template from the selected concept and permanently deletes the template's associated data from the database.
- **Delete:** Deletes the templates and all associated data from the database.
- **Refresh:** Updates the template list.
- **Rename:** Renames the template.

To Open a Template

Double-click or drag and drop a template into the workspace to open the template in a Template Wizard editor.

9.8.2   Exposed Objects
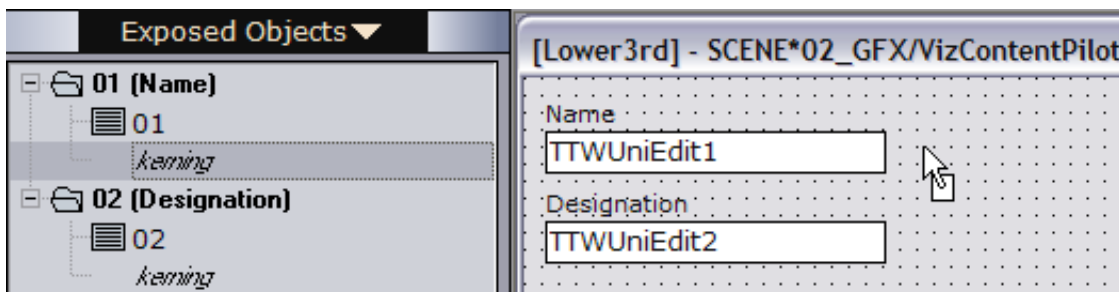
The **Exposed Objects** view displays a selected template's exposed objects. Like the older style Scene Objects template, exposed objects can be dragged and dropped into the template for further customization.

With a template loaded, the **Exposed Objects** window will show a list of all exposed objects and properties of a scene.

Selecting a component on the template highlights the object it refers to. If an object's property has been selected from the Exposed objects list, the corresponding component will also be highlighted on the template. In both cases the Template Wizard Components opens, displaying the component's properties. The object **Name** has one component on the template: a text box (01) for the Name object. A second text box component can be added for kerning (01.kerning).

> ⚠ **Note:** A template with exposed objects uses control plugins.

Adding a Component from the Exposed Objects Pane

Drag and drop the exposed object onto the template and a default component will be added based on the data type of the property.

9.8.3   Scene Objects

The Scene Objects view displays the scene tree together with its scene objects, available control channels and directors. As with an Exposed Objects template, scene objects can be dragged and dropped onto the template for further customization.

> ⚠️ **Note:** Exposing scene objects is commonly performed with Viz Pilot version 4 and older as these versions do not support control plugins.

9.8.4   Layer Panel
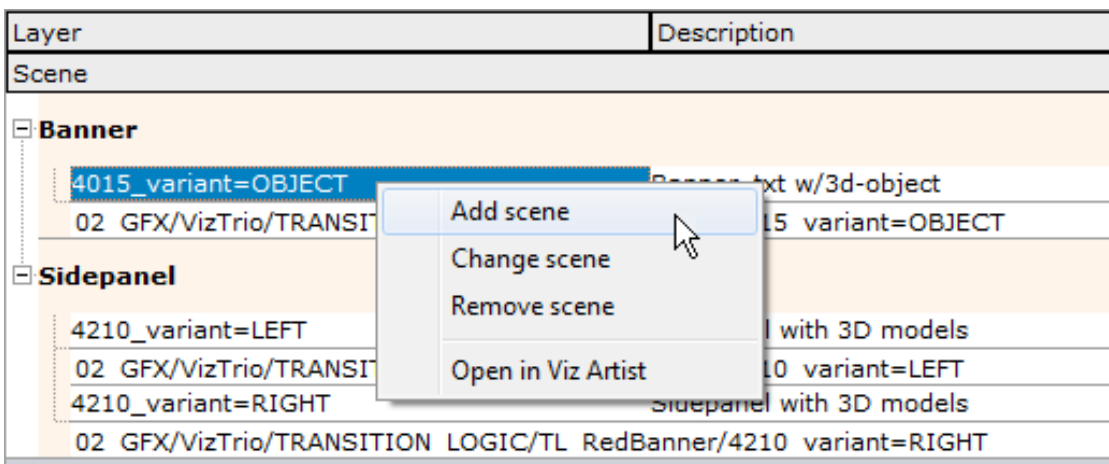
When creating templates based on transition logic scenes, the layer panel is displayed (lower-left). The layer panel is shown when editing transition logic templates, and enables the template designer to see the logical layers of the template and all front scenes within a specific layer. The panel also allows the user to change the description of the front scene, and open it in Viz Artist for editing to add, change or remove a front scene.

In a template, each layer is represented with a drop-down menu with all front scenes that can be taken on-air for that specific layer.

- **Add scene:** Adds a front scene to the template either as part of an existing layer or as part of a new layer. Additional front scenes must refer to the same background scene as the other scenes.
- **Change scene:** Changes the front scene in the selected layer.
- **Remove scene:** Removes the front scene from the selected layer.

- **Open in Viz Artist:** Opens the front scene in Viz Artist for editing.

⚠️ **Note:** Transition Logic templates that reside in two or more concepts are kept synchronized when adding or changing scenes. Concepts that do not contain the same set of front scenes on Viz or have a different background scene, are not in sync. If this occurs, the operation is canceled and there will be information on the problem.
Out of sync situations may occur if a transition logic template that resides in two or more concepts is opened, and front scenes only finds one of the concept's scene paths or a scene with a different background scene is changed or added.

9.9   Viz Pilot News Test Page

This web page loads Viz Pilot News within Internet Explorer (IE) so that it can be used for testing Viz Pilot's newsroom connection properties and functionality.

Open the test page from the **Start Menu > Viz Pilot News Test Page**.
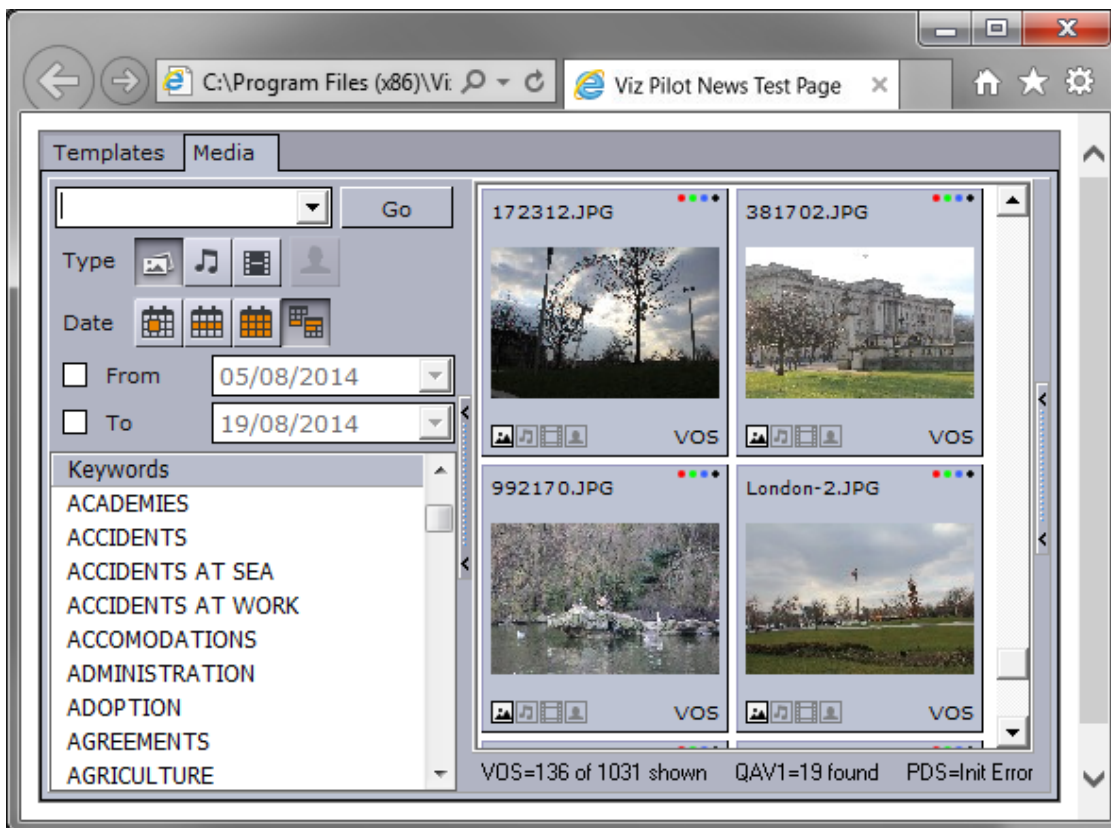
To edit the connection settings, reinstall Viz Pilot.

> ⚠️ **Note:** When the Viz Pilot News Test Page is opened, IE may issue a warning and ask the user to allow it to run within IE. To allow the test page to run without this warning, open IE, and go to **Tools > Internet Options > Advanced > Security**, and then check the option called Allow active content to run in files on My Computer.

9.10 Update Script Editor

The Update Script Editor lets you add scripts that will be executed by update services on the server side, and not on the client side through Director.

Unlike the Live Update mechanism that relies on Director executing the scripts, update services work on the Media Sequencer alone, and are therefore available to all products connected to the Media Sequencer.

The Update Script Editor does not have access to components in the template, only the data, so the template does not need to be loaded by Director (as is the case for Live Update scripts).

This section contains the following topics:

- Using the Update Script Editor
- Update Services
- Update Service Scripting
- Viz Data Format (VDF)

9.10.1 Using the Update Script Editor

The Update Script Editor consists of the Script and Actions tabs.

Script
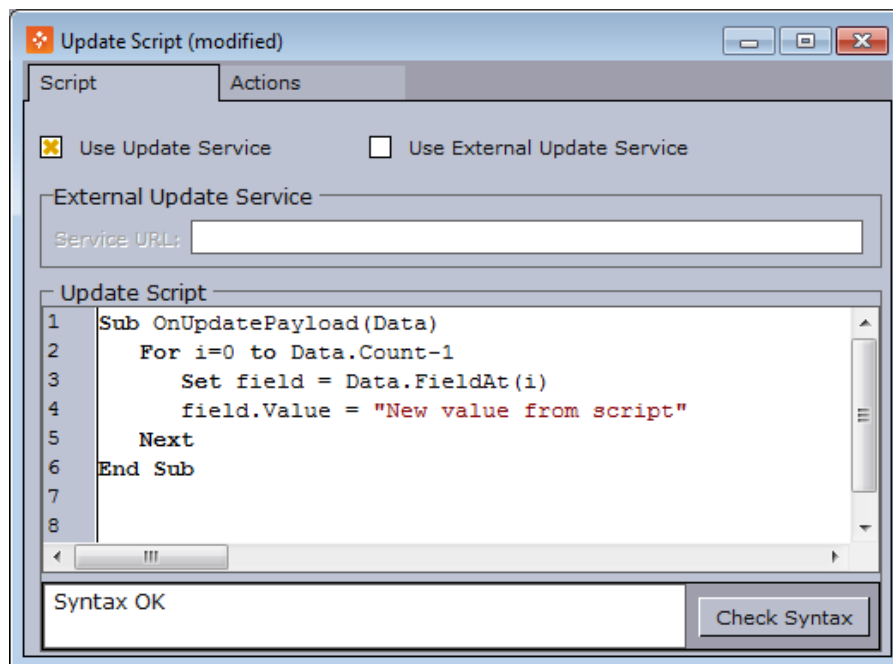
The **Script** tab lets you select which of the Update Services you want to use. Either Viz Pilot system's own Update Service or an external (third party) update service.

Actions
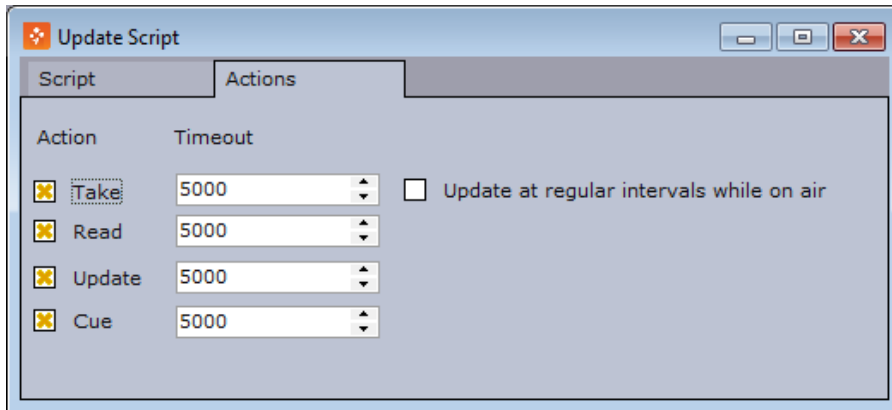
The **Actions** tab lets you select which actions are triggered during an update and set the timeout for each action.

Script actions are based on **Read**, **Take**, **Cue** and **Update** events that are triggered on Media Sequencer.

When the **Take** action is selected, the **Update at regular intervals while on air** option is enabled. If this is checked, Media Sequencer will periodically send updates to the update service. The interval is specified by the parameter `live_update_interval`, and the minimum value is 2 seconds.

⚠️ **Note: Update at regular intervals while on air** is only available if Template Wizard is connected to a database with schema Viz Pilot 6.0 or later, and Media Sequencer 3.0 or later.

9.10.2   Update Services

Update services are mechanisms that let the template designer decide when to update data in the template, and which fields to update. This is useful for data that must be updated right before going on air. Typical examples are stock prices, player statistics etc.
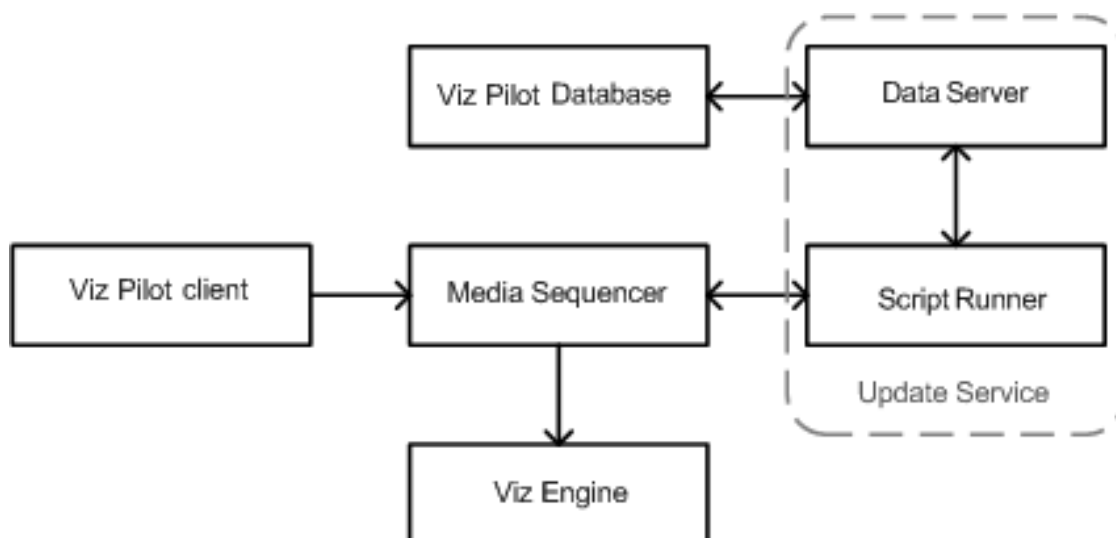
In Template Wizard you have two options:

1. **Viz Pilot's Update Service (Script Runner):** This mechanism lets you write a script to update the data that will be run by Script Runner during playout.
2. **External Update Service:** Alternatively, you can point the template (no script) to an external update service (for example third party), that can return updated data using the Viz Data Format (VDF). In order to implement an external update service, you need to parse and return a VDF document to the Media Sequencer. The document is posted to the external update service by the Media Sequencer.
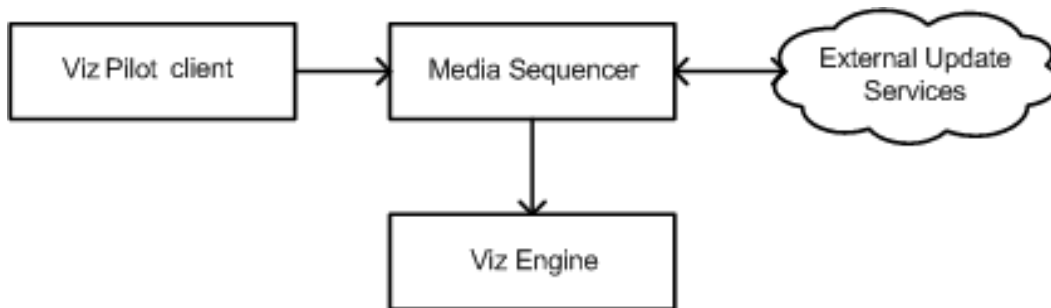
9.10.3   Update Service Scripting

The entry point for using Viz Pilot's update service (Script Runner) is a method called `OnUpdatePayload`. The method has the following signature: `Sub OnUpdatePayload(Data)`.

The `Data` parameter contains the incoming data and the data can be modified in the `OnUpdatePayload` script.

Properties and Methods

Data Properties	Description
Data.Count	Number of fields in the Data.
Data.FieldAt(index)	Returns a PayloadField at the given index.
Data("1")	Returns the PayloadField with the given name. "1" in this example.
PayloadField properties	**Description**
Name	The name of the field.
Value	The name of the field.

Example

A sample script that updates all the incoming fields and set their values to "New value from script":

```
Sub OnUpdatePayload(Data)
For i=0 to Data.Count-1
    Set field = Data.FieldAt(i)
    field.Value = "New value from script"
Next
End Sub
```

> ✅ **Tip:** Data treated outside the `OnUpdatePayload` function (for example fetching external data from files or databases) must be returned before the `OnUpdatePayload` function ends.

9.10.4　Viz Data Format (VDF)

Viz Data Format (VDF) refers to format specifications created for the purpose of data interchange between Vizrt software components. VDF forms the basis of integration points, interoperability, reusable components and content portability.

The *Viz Data Format (VDF) Payload and Model Specification* can be accessed in the Web Interface section of the Viz Pilot Data Server Administrator Guide.

Example VDF

```
<?xml version="1.0" encoding="utf-8"?>
<payload model="http://localhost:8177/templates/18110/model" xmlns="http://
www.vizrt.com/types">
    <field name="-concept-variant-choice">
        <field name="concept">
            <value>Demo</value>
        </field>
        <field name="variant">
            <value>Default</value>
        </field>
    </field>
    <field name="1">
        <value>Pilot</value>
    </field>
    <field name="2">
        <value>rocks :)</value>
    </field>
</payload>
```

9.11 Execution Logic Editor

Execution Logic commands are saved as part of a template. This allows data elements based on the template to use the same execution logic commands.

For example, adding an execution logic script to the Media Sequencer command **Take**, will replace the **Take** command for all data elements based on that template.

A benefit of using Execution Logic is that the script can be run without the need for Director to be open. It is possible to utilize a limited set of commands - or any Viz Engine command - directly on the Media Sequencer to issue instructions or emulate button presses for each created Data Element.

This section contains the following topics:

- Execution Logic Editor
    - Enabling the Execution Logic Editor
    - Adding Execution Logic to a Template
- Working with Execution Logic
    - Sending Basic Commands
    - Example - Play, Continue, Take Out
    - Example - Forked Execution
    - Example - Commands Generated by a Template

9.11.1 Execution Logic Editor
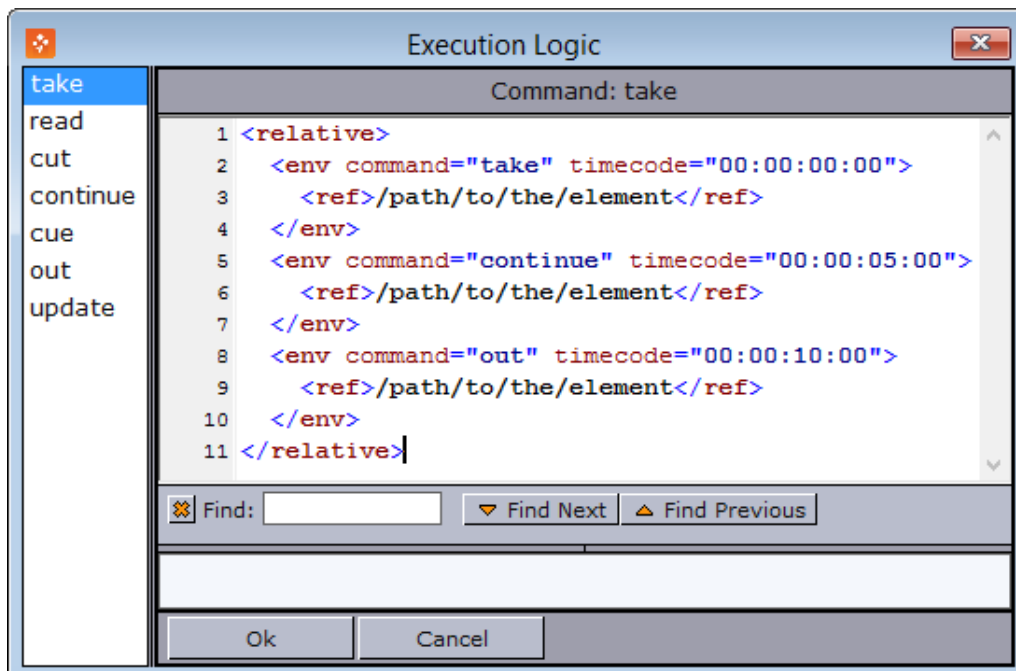
The Execution Logic editor consists of three panes: a list of commands, a command editor and a list-view displaying syntax errors. The editor also has a search feature.

- **Command list**: Displays the available commands. Available context menu options are **Clear selected command** and **Clear all commands**.
    - **Clear selected command**: Clears the execution logic for the selected command.
    - **Clear all commands**: Clears all execution logic for all commands.
- **Command editor**: Displays the currently selected command and its execution logic. Available context menu options are **Insert default action** and **Find**.
    - **Insert default action**: Inserts the default command `<ref>/logic/element/run</ref>`.
    - **Find (CTRL + F)**: Opens the **Find** tool.
- **Syntax error list**: Displays any syntax errors in any of the commands. It is not possible to close the window with changes to any command before all syntax errors are resolved.
- **Find (CTRL + F)**: Searches the execution logic of the command that is currently open.
- **OK**: Saves changes and closes the execution logic editor.

> ⊗ **IMPORTANT!** If the Execution Logic Editor is closed using the **Exit** button or by selecting **File** then **Close**, any changes will NOT be saved. In addition, if the template is saved and closed without first saving the execution logic, the execution logic will not be updated in the template.

- **Cancel**: Close without saving.
- **XML tags suggestion dropdown**: Type **<** to open the suggestion dropdown menu.
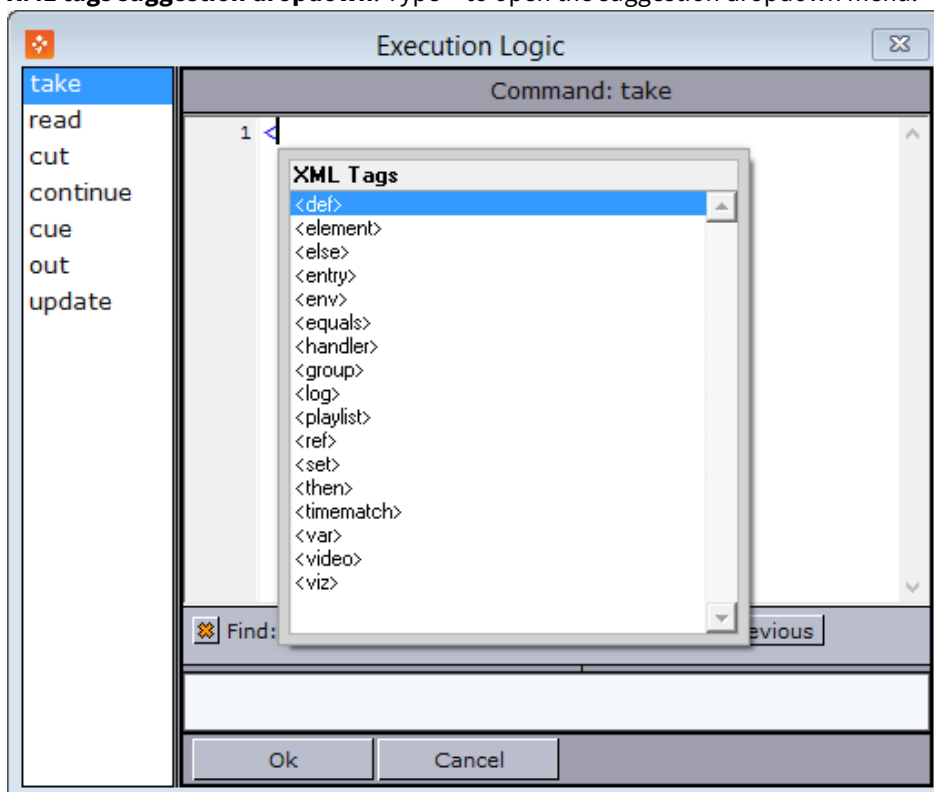
Enabling the Execution Logic Editor

Go to **View** > **Execution Logic Editor**.

Adding Execution Logic to a Template

1. Open the Execution Logic Editor.
2. Click on one of the existing commands (left pane).
3. Add the logic in the editor (right pane) and click **OK**.

> ⚠️ **Note:** For more information on how to create execution logic, see the Media Sequencer manual on how to use the Media Sequencer's logic building blocks.

9.11.2 Working with Execution Logic

Sending Basic Commands

The `<viz>` handler is used to send commands to the Viz Engine.

The following example sends the `RENDERER SET_OBJECT SCENE*...` command to channel "A" in the current profile:

```
<env viz="A">
    <viz>RENDERER SET_OBJECT SCENE*...</viz>
</env>
```

Multiple commands can be sent by separating each command with `<br/>`, for example:

```
<viz>RENDERER SET_OBJECT ...<br/>RENDERER*STAGE START</viz>
```

Instead of setting the command directly, a more powerful approach is to use the contents of a field in the template. The field (a hidden textbox for instance) can then be filled with the Viz commands that need to be sent.

The example below shows how the contents of a data field in a data element can be retrieved by using the `<ref>` construct (`"field_01"` is the ControlObjectName of the data field):

```
<env viz="A">
    <ref><var>element</var>/data/field_01</ref>
</env>
```

To send commands to several channels, duplicate the command:

```
<env viz="A">
    <ref><var>element</var>/data/field_01</ref>
</env>
<env viz="B">
    <ref><var>element</var>/data/field_02</ref>
```

```
</env>
```

⚠️ **Note:** Forked execution is required when a channel contains multiple engines, otherwise the commands will only apply to the first engine in a channel.

⚠️ **Note:** Execution logic will not work with non-control object based templates.

Example - Play, Continue, Take Out

This example shows how execution logic can be used to play an element, do a *Continue* after five (5) seconds, and then a *Take Out* after ten (10) seconds.

In the Execution Logic Editor, select the **take** command. Add the logic into the editor (right pane). This means that when a "take" is issued on a data element based on this template, the Media Sequencer will execute the logic.

The commands are modified to do a `"take",` `"continue"` and then an `"out"` . The timecode for each operation must be set.

```
<relative>
    <env command="take" timecode="00:00:00:00">
        <ref>/logic/element/run</ref>
    </env>
    <env command="continue" timecode="00:00:05:00">
        <ref>/logic/element/run</ref>
    </env>
    <env command="out" timecode="00:00:10:00">
        <ref>/logic/element/run</ref>
    </env>
</relative>
```

⚠️ **Note:** The running/outer context takes precedence over attributes on the "ref". Instead of adding attributes on the "ref" node, use an "env" node as in the example above.

Example - Forked Execution

These examples show how the "take" command can be modified to make the template override the standard logic and instead send `RENDERER*STAGE START` .

Below, the command will be sent to the channel assigned to the data element:

```
<forked_exec>
    <entry name="execution_group"><var>channel</var></entry>
        <viz>RENDERER*STAGE START</viz>
</forked_exec>
```

To send commands to a specific channel in the current profile, replace `<var>channel</var>` with the name of the channel you want to send to, as follows:

```
<forked_exec>
    <entry name="execution_group">MY_CHANNEL</entry>
        <viz>RENDERER*STAGE START</viz>
</forked_exec>
```

Example - Commands Generated by a Template

By using the information from the preceding examples, we can create logic that will send custom Viz commands that are generated by the template.

One way is to add a Value Control Component to the template, and set the ControlObjectName to `"vizcmds"`. Next, create a regular script that sets the ControlValue of the TWValueControl to whatever command needs to be sent.

Alternatively, use a standard memo box and set the ControlObjectName to `"vizcmds"`. Next, enter the Viz commands (or script what the contents are to be). The memo box's visibility can be set to false so that it will not be visible to the user. In the Execution Logic you can then add the following:

```
<forked_exec>
    <entry name="execution_group"><var>channel</var></entry>
    <viz>
        <ref><var>element</var>/data/vizcmds</ref>
    </viz>
</forked_exec>
```

9.12 Getting Started

Template Wizard is a stand-alone application that can be started from the program menu, a desktop shortcut or from Director if it is installed on the same machine.

> ⚠️ **Note:** Some Template Wizard settings are configurable, these settings are stored in the *ContentPilot.ini* file. See the section on Initialization and Configuration Files, and information about ContentPilot.ini parameters and how to edit them using the Initialization File Editor.

This section contains the following topics:

- Starting Template Wizard
- Specifying Command Line Parameters for Template Wizard
- Starting Template Wizard from Director

9.12.1 Starting Template Wizard

Open the program from the Windows Start Menu (**Template Wizard**), or the desktop icon .

9.12.2 Specifying Command Line Parameters for Template Wizard

1. Right-click the Template Wizard desktop shortcut, and select **Properties**.
2. In the Properties window select the **Shortcut** tab.
3. To start multiple instances of Template Wizard add `-duplicates` to the **Target** path.

   - `"%ProgramFiles(x86)%\vizrt\Viz Pilot x.y\VizTemplateWizard.exe" -duplicates`

     > ⚠️ **Note:** When starting a duplicate instance you will most probably receive a socket error. Confirm it and you will be able to run duplicate instances of Template Wizard.

4. To specify the Media Sequencer to connect to, add `-mse <host>` to the **Target** path.

   - `"%ProgramFiles(x86)%\vizrt\Viz Pilot x.y\VizTemplateWizard.exe" -mse <host>`

5. To specify the database to connect to, add `-db <connect string>` to the **Target** path.

   - `"%ProgramFiles(x86)%\vizrt\Viz Pilot x.y\VizTemplateWizard.exe" -db <connect string>`

6. To specify the inifile to use, add `-inifile <inifile>` to the **Target** path.

   - `"%ProgramFiles(x86)%\vizrt\Viz Pilot x.y\VizTemplateWizard.exe" -inifile <inifile>`

9.12.3   Starting Template Wizard from Director

1. Select **Template Wizard** on the **Tools** menu, or
2. Right-click a template in the template list, and select **Edit Template** on the context menu.
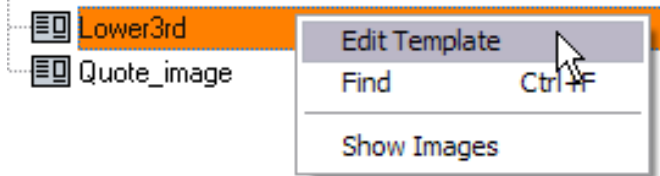
9.13   Main Menu

Template Wizard has six Main Menu options, as described below:

- File
- Edit
- View
- Viz Engine
- Tools
- Help
    - Save Template
- Save Template to File
- Change Scene
    - Changing a Scene for Design
    - Changing a Scene for Playout
    - Converting a Template Using a Control Object-based Scene
- Selecting a Viz Engine
- Refreshing Scene Information
    - Refreshing Scene Information
- Script Repository Browser

9.13.1   File

The File menu has the following options:

- **New (CTRL + N)**: Creates a new blank template.
- **Load**...: Opens a previous template made in Template Wizard.
- **Load from file** ...: Opens a template from file.
- **Save (CTRL + S)**...: saves the template to the database.
- **Save to File**...: Is used for saving a template to a file, for exporting to another database, to use in Viz Trio or to have a backup. See Save Template to File.
- **Close**: Closes the current template.
- **Wizard**...: Opens the Wizard to select scenes and exposed scene objects for template creation.
- **Use Script Unit** ...: Includes a script unit in a template script.
- **Refresh Scripts**: Loads updated script units from the database.
- **Reconnect to database**: If the database connection is lost, this option tries to reconnect to the template database.
- **Exit**: Exits the Template Wizard program.

9.13.2   Edit

The Edit menu has the following options:

- **Change Scene:** Use this option to Change Scene for the template's scene in design mode. To change scenes for templates used during playout, the **Scene name** property set by the Template Manager in the Variants pane must point to a different scene.

> ⚠ **Note:** Change Scene is not supported for transition logic templates.

- **Undo (CTRL + Z)**: Undoes the latest change.
- **Redo (CTRL + Y)**: Redoes the latest undo.
- **Select All (CTRL + A)**: Selects all components in the template.
- **Bring to Front**: Moves an object on the template to front.
- **Send to Back**: Moves an object on the template into the background.

9.13.3   View

The View menu has the following options:

- **Command Editor**: Shows the command editor.
- **Script Editor**: Opens the Script Editor.
- **Update Script Editor**: Opens the Update Script Editor.
- **Execution Logic Editor**: Opens the Execution Logic Editor.
- **Snap to Grid**: Enables template components, when added or moved, to automatically adjust (snap) itself to the template's grid.

9.13.4   Viz Engine

The Viz Engine menu has the following options:

- **Select Viz Engine**: Select Viz Engine to connect and render to.
- **Set On Air**: Sets the connected Viz Engine renderer in On-Air mode.
- **Show Bounding Box**: Shows the boundaries of the scene's objects when previewing.
- **Show Commands On**: Displays the **Show commands** console on the renderer.
- **Show Commands Off**: Closes the **Show commands** console on the renderer.
- **Cleanup Renderer**: Clears all scenes loaded on the renderer.
- **Remote Preview (CTRL + P)**: Launches the remote Remote Preview in Template Wizard window. This option requires a remote Viz Engine to be in On-air mode.
- **Local Viz Preview**: Launches the Local Preview in Template Wizard window. This option also starts the Viz Engine instance.

9.13.5   Tools

The Tools menu contains the following options:

- **Categories & Channels**...: Opens the Categories and Channels window to add categories and channels and define a mapping between them. Templates can then be added to a category, and thereby be assigned to a default output channel.
- **Customize Keyboard**...: Opens the Keyboard Configuration window.
- **Show Command Console**: Opens the Template Wizard Macro Commands window.

- **Refresh Scene Info**: Enables the user to Refresh Scene Information if the scene information has changed after the template was created or updated the last time. This updates the template with changes to control objects and background scenes.

> ⚠️ **Note:** Since existing data elements refer to the background scene in the template, these will also change. i.e. the background scene will be updated for the template and all it's data elements every time this menu option is selected.

- **Component Palette**: Opens the Component Palette, enabling the user to select which components to be displayed in the drop-list and on the toolbar.
- **Script Repository**: Shows all script units in the Script Repository Browser (database), and which templates and data elements that are using it.
- **Template Manager**...: Opens the Template Manager window.
- **Export:** Export templates to an archive file (see Export and Import in Template Wizard).
- **Import**: Import templates from an archive file.

9.13.6   Help

- **Template Wizard Help (F1)**: Opens the online version of the *Viz Pilot User Guide* (this document). A local copy can be accessed from the Windows Start Menu.
- **About**: Opens the About window, that contains information about the installed Viz Pilot version.

Save Template
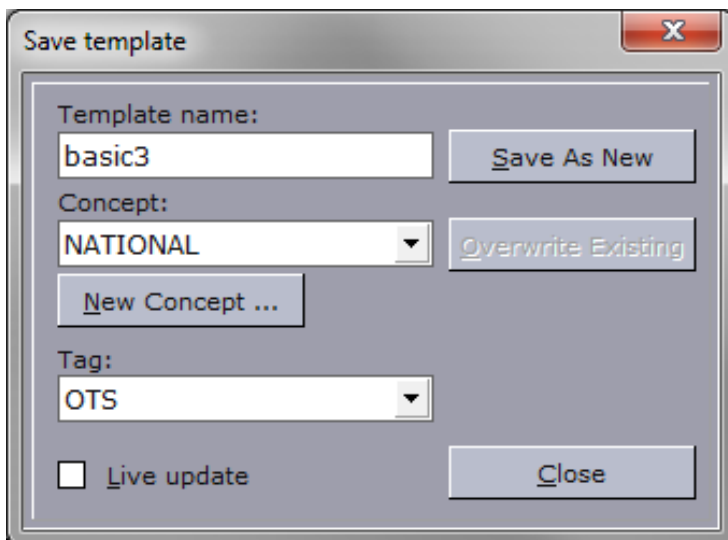
- **Template Name**: Sets the template name for the template.
- **Concept**: Sets the concept name for the template.
- **New Concept**...: Opens the New Concept dialog box for creating new concepts.
- **Tag**: Allows the template to be tagged, by selecting from a user-defined list. (See Tags).
- **Live Update**: Enables the template to use Live Update for last-minute updates before playout (for example weather or stock updates). This requires template scripting.

> ⚠ **Note:** Live Update has been superseded by Update Services. If you need to use Live Update, you must also use the Legacy Playlist.

- **Save As New (ALT + S)**: Saves the template and closes the dialog.
- **Overwrite Existing (ALT + O)**: Overwrites the existing template with the existing name or a new name.
- **Close**: Closes the save dialog without saving the template.

> ⚠ **Note:** It's possible to save a template created in Template Wizard and run it as a graphical user interface, including scripts, either:
>   - inside Viz Trio (called "Active TW template") or,
>   - inside Ticker Client, instead of the default message editor (called "presentation file").

9.13.7   Save Template to File

Save to File is used for saving a template to a file, for exporting it to another database, for using a file in Viz Trio or for making a backup of the file.

It's important to select the correct extension when saving:

- `*.vtw` : Template Wizard template. Used for normal templates.
- `*.vtwc` : Template Wizard control template. Used for templates that are based on a scene with control object plugins.

    When saving a `.vtwc` control template, a `*.vtwc.controlpage` file is also saved, which contains the description of the scene that the template controls (the internal TrioPage instance). Both files should be kept, as both are required if the template is to be loaded from file in future.

> ❌ **IMPORTANT!** Save to File does not preserve Tags or Variants information or Script Units. To preserve these values, use Export and Import in Template Wizard.

9.13.8   Change Scene

Change scene is used to change the scene a template is referencing while designing.

Templates that are not control object based (for example templates built in Template Wizard versions prior to Viz Pilot 5) can change the scene and convert the template to use control objects if the scene has been rebuilt using control object.

> ⚠ **Note:** Transition logic scenes with control object cannot change scenes.

Conversion is not required if a template is created without referencing a scene or if it is based on a scene that does not use control object. Conversion can only take place if the new scene uses the control object plugin.

Converted templates show exposed objects as a tree structure with nodes, and not the complete Viz Artist scene tree with containers.

Changing a Scene for Design

Click **Change Scene** on the **Edit** menu, and add the new scene using the Wizard.

Changing a Scene for Playout

1. Start Template Manager.
2. Select the template variant to edit and click **Edit**.
3. Set the new scene and click **Save**.

Converting a Template Using a Control Object-based Scene
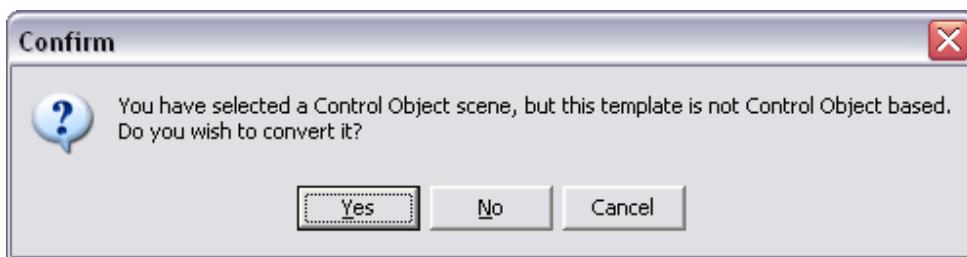
1. Select the **Change Scene** option on the **Edit** menu.
2. Select a scene using the Wizard.
3. Confirm if the template should be converted to a control object based template.

> ⚠ **Note:** Changing a scene without changing the scene name in Template Manager under the Variants section will only change the scene used during design.

9.13.9 Selecting a Viz Engine
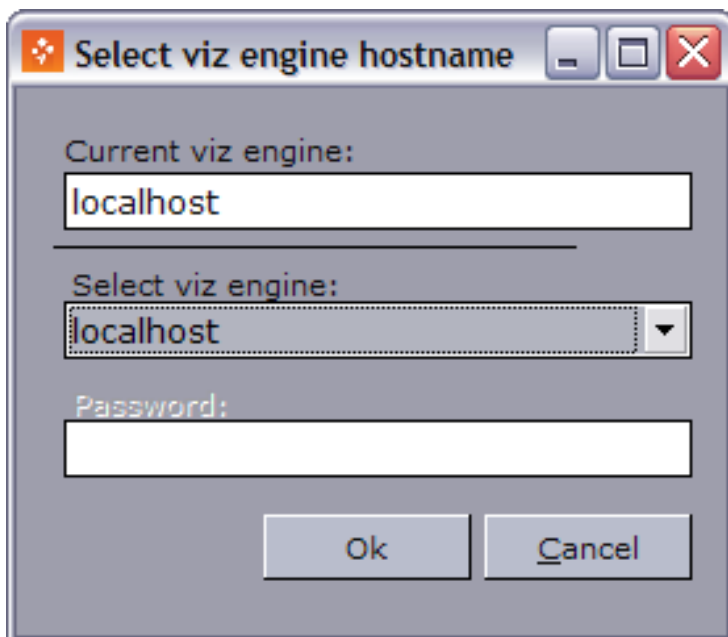
> ⚠️ **Note:** Machines that are not shown in the list must be added during installation or to the _ContentPilot.ini _file under the SGI section.

Select **Select Viz Engine** on the Viz Engine menu, and from the dialog box select the Viz to use, and click **Ok**.

> ⚠️ **Note:** Configuration is done in the `ContentPilot.ini` initialization file.

9.13.10   Refreshing Scene Information
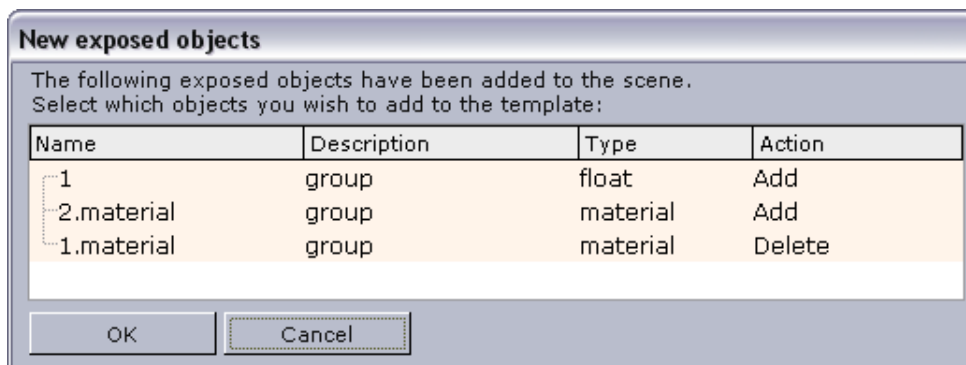
When editing a template, it is possible to edit a scene in Viz Artist and make changes to the template's scene. Any changes made to the scene can be updated directly by the template designer.

The template will be updated when the field identifier for a specific component has been changed through a new component being added to the template. However, the old component with the old field identifier will still be present. The Exposed Objects list is updated correctly, therefore, it is safe to delete any unused components.

Refreshing Scene Information

1. Start Viz Artist.
2. Open the template's scene, edit and save the changes.
3. Open Template Wizard, and click Main Menu **> Tools > Refresh Scene Info**.

> ⚠️ **Note:** Clicking **Refresh Scene Info** updates the template with changes to control objects and background scenes. Since existing data elements refer to the background scene in the template, these will also change. The background scene will be updated for the template and all of its data elements every time this menu option is selected.

4. Click **OK** to add or delete exposed objects.
5. *Optional:* Remove any unused components that may have been made obsolete by the update.

9.13.11   Script Repository Browser
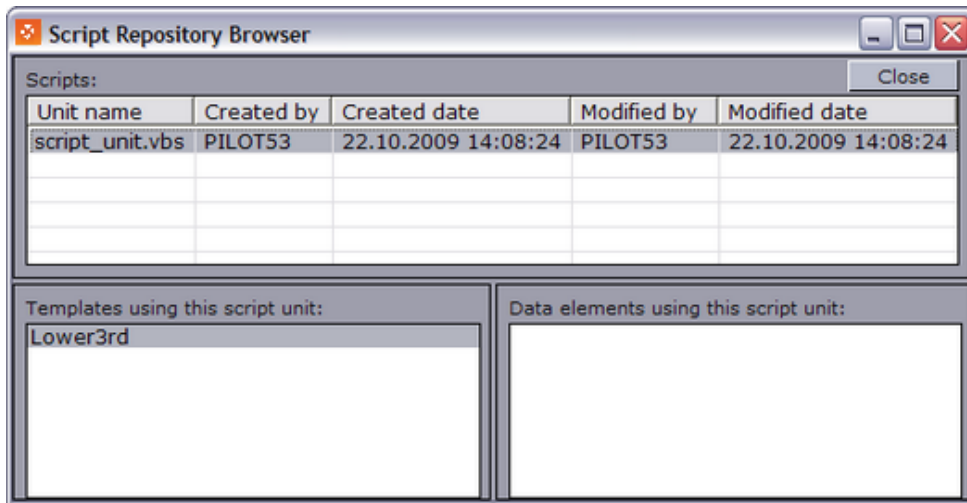
This panel provides an overview of available scripts, the templates and data elements that are using the scripts. It is also possible to delete script units from the repository browser, but this affects the templates that use the scripts. A warning appears if the script is used by a template.

9.14   Script Editor
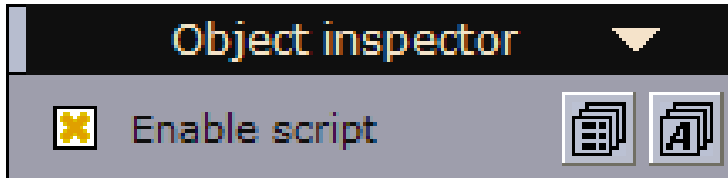
There are two ways to add scripts to a template. Scripts that are specific to the template can be added in the Main script tab, or as separate scripts (from the Main script) created as Script Units.

> ✅ **Tip:** Script Units can be stored as reusable scripts for more than one template and can be used as a means of organizing scripts into functional and logical pieces.

This section contains the following topics:

- Context Menu
- Script Editor Toolbar
- Script Units
  - Inserting a Local Script into the Main Script
  - Using a Script Unit from the Repository
  - Reopening Used Script Units
  - Saving Script Units to the Repository
  - Retrieving Script Units From the Database
  - Resolving Conflicts Between Local and Repository Script Units
  - Exporting a Template That Uses Script Units
- Live Update
- Viz Trio Template
  - Script Example
  - Creating a Template for Viz Trio

9.14.1   Context Menu
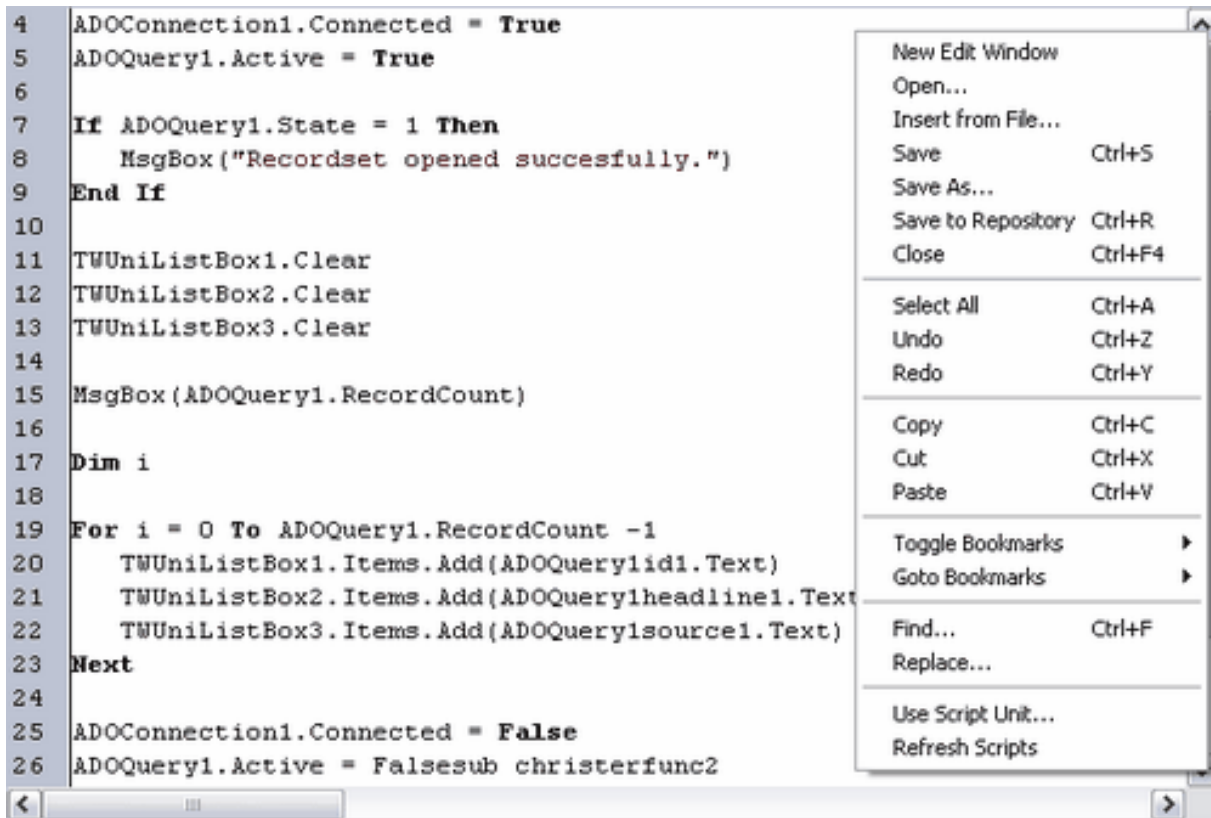
- **New edit window:** Opens a tab with a new script editor.
- **Open:** Opens a script file in a separate tab. See also Reopen used script units.
- **Insert from file**...: Inserts script file, adding it as a script unit to the script that was open at the time of the operation.
- **Save (CTRL + S):** Saves a script unit to the computer (not to repository).
- **Save As:** Saves a script unit with a new name to the computer (not to repository).
- **Save to repository (CTRL + R):** Saves edited script units to the database repository.

> ⚠ **Note:** When a template with script units is saved, a dialog appears asking for script units to be saved to the existing database. Use the **Save to Repository** menu option to save script units to the database without saving the template.

- **Close (CTRL + F4):** Closes a script unit.

> ⚠ **Note:** The options **Save**, **Save As**, **Save to Repository** and **Close** are only available for Unit scripting windows and are therefore disabled for the Main scripting window.

- **Select All (CTRL + A):** Selects all text in the script.
- **Undo (CTRL + Z):** Undoes the latest change.
- **Redo (CTRL + Y):** Redoes the latest undo action.

- **Copy (CTRL + C):** Copies selected text in the script.
- **Cut (CTRL + X):** Cuts selected text in the script.
- **Paste (CTRL + V):** Pastes content from the clipboard.
- **Toggle Bookmarks:** Adds and remove a bookmark in the script at the line where the cursor is by toggling bookmarks on and off.
- **Goto Bookmarks:** Makes the cursor jump to a bookmark.
- **Find… (CTRL + F):** Searches for a text string.
- **Replace:** Searches for a text string and replace it by another string.
- **Use script unit**..: Makes the main script use a script unit. A string of the form: `'USEUNIT <myscript>`, in the main script shows that a unit is used. The script units are saved back to the repository when a template is saved and the local copies are updated every time the program is started. Script units are often created to be generic and reusable.
- **Refresh Scripts:** Loads updated script units from the database.

9.14.2   Script Editor Toolbar

- **Open Script Unit:** Opens a script file in a separate tab. See also Reopen used script units.
- **Predefined Functions:** Shows the predefined functions and variables that are available, together with a brief description. Use the **Add** button to add the selected function to the script.
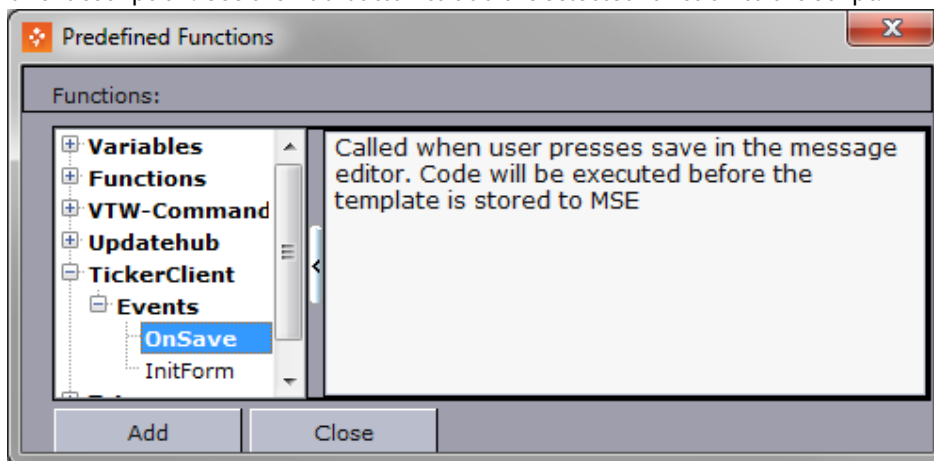
- **Show Library Scripts:** Displays available Script Units that are not already in use as script units in the currently open script.

> ⚠️ **Note:** Scripts added to a local folder that are to be in permanent use should be added to the database repository, see Save script units to the repository.

- **Syntax Check:** This button checks the syntax of the scripts and reports errors. Errors are shown below the script pane.

- **Save to Repository (CTRL + R):** Saves edited script units to the database repository. For details, see Save script units to the repository.
- **Close Unit (CTRL + F4):** Closes the selected script unit. See also how to Reopen used script units.
- **Change Font:** Change the font type and font size for the script editor.
- **Drop-list:** The drop-list displays all functions and procedures that are declared in the script. Select an item from the list to jump to the point in the script where that function or procedure starts.

9.14.3   Script Units

All script units are stored on the database. When Template Wizard is started, all script units from the database repository are downloaded to a local folder.

- `%PROGRAMDATA%\Vizrt\Viz Pilot x.y\Scripts`

  New script units can be saved to the database when the template is saved. If the script unit is updated and already exists in the database, a dialog with three options appears asking if the database script unit should be updated with the local copy. The three options are:
    - **Yes:** Saves the template, and updates the script units on the database.
    - **No:** Saves the template, but does not update the script units on the database.
    - **Cancel:** Cancels all save operations for the template and script units.
      Script units that are saved back to the repository when a template is saved updates the local script unit copies every time the program is started.

Inserting a Local Script into the Main Script

1. Place the marker in the current script where the contents of the new script are to be placed.
2. Right-click the script unit and on the Context Menu select **Insert from File …** .
3. In the **Open** dialog box, select the script and click **Open**.

Using a Script Unit from the Repository

1. Click the **Show library scripts** button to open the **Use Unit** dialog box.
2. Search for and select the script unit to add, and click **OK**.
    - A new tab will open in the **Script Editor**, showing the script unit.
    - A string of the form `'USEUNIT <myscript>` will be added into the main script to indicate that the selected script unit is being used.

> ✅ **Tip:** To stop using a script unit, delete the relevant `'USEUNIT` comment from the main script.

Reopening Used Script Units

By default, a template with script units will not display the script unit tabs if the template is closed and later reopened. To reopen used script units do either of the following:

- Place the marker in front of the `'USEUNIT <myscript>` tag and press **CTRL + ENTER**,

  or
- Right-click the script unit and on the Context Menu that appears, select **Open ...** .
- In the **Open** dialog box, select the used script unit and click **Open**.

Saving Script Units to the Repository
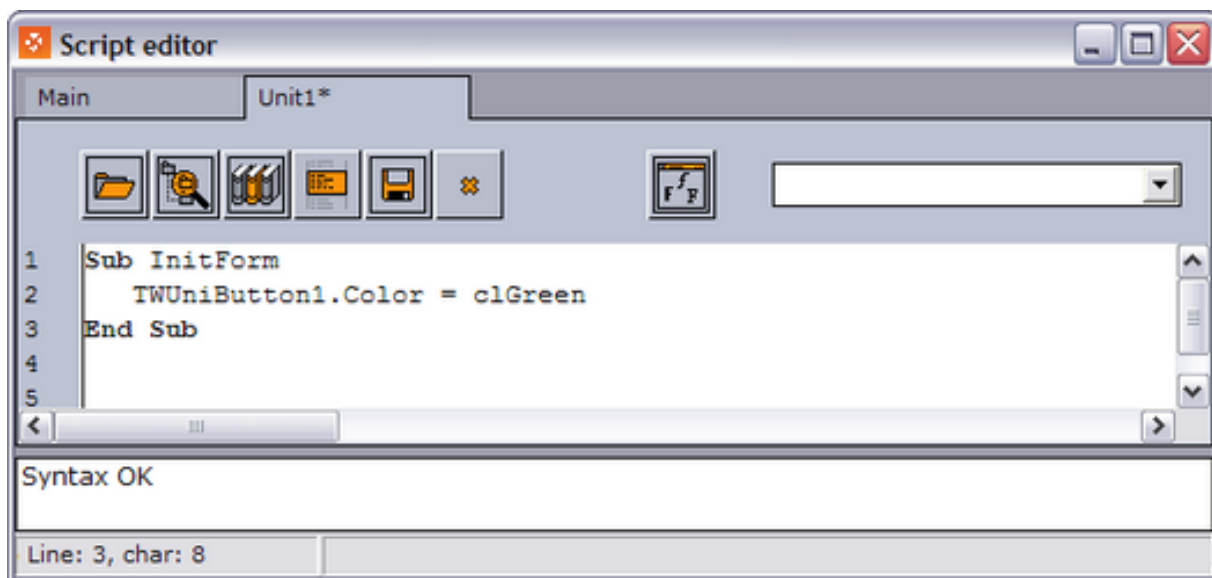
1. Open the Script Editor.
2. Right-click the editor, and on the Context Menu select **New Edit Window**.
3. Write/Add the script to save to the repository.
4. Click the Save button (**CTRL + R**).
5. In the **Save As** dialog box enter the File name and click **Save**.

> ✓ **Tip:** Create and save a test script and delete it from the local Script Units folder. Restart Template Wizard to test that the script has been uploaded correctly.

Retrieving Script Units From the Database

This procedure can be used to download new or updated scripts while Template Wizard is running (for example, on other clients).

On the **File** menu, select **Refresh Scripts**.

Resolving Conflicts Between Local and Repository Script Units
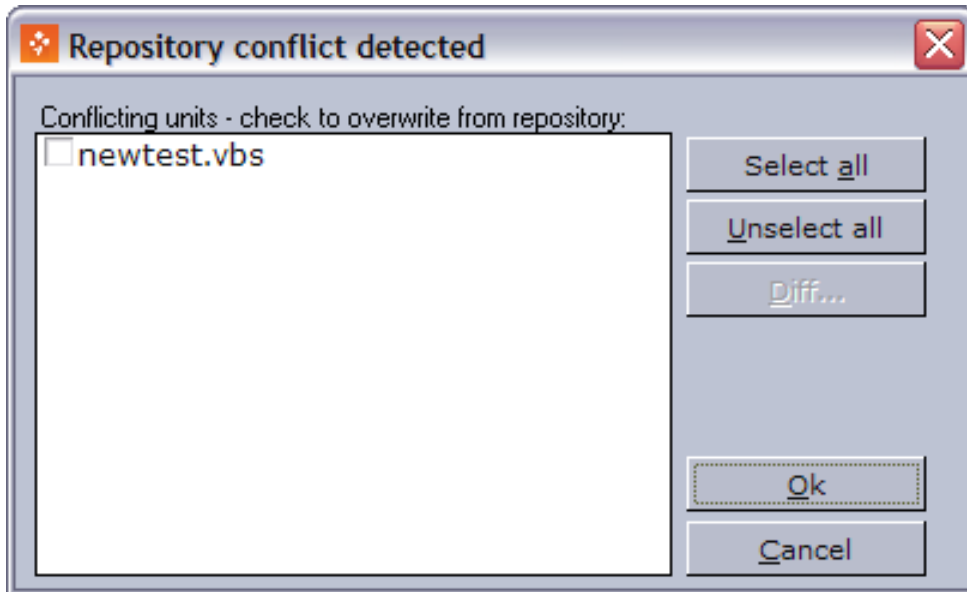
When new files are uploaded to the client, there might be cases where repository files are different from local files. To handle situations like these, a user can do one of two things:

- **Overwrite** the local file with the repository file,
  or
- **Rename** the local file and then overwrite.
  It is also possible to use a file comparison tool (diff tool) to compare the file in the repository with your local file. When specifying which tool to use, add the entry `diff_tool` in the *ContentPilot.ini* file. The `diff_tool` entry must also be followed by the path to the tool itself.

> ℹ️ **Example:** *diff_tool*=C:\ProgramFiles\Araxis\Araxis Merge 2001 v6.0 Professional\Merge.exe

Exporting a Template That Uses Script Units

To export a template that uses script units, use the export/import functionality in Template Wizard, see Export and Import in Template Wizard.

Save Template to File can be used to save templates, but it does not save the included Script Units. If using **Save to File**, the script unit files must be copied separately.

9.14.4   Live Update

> ❌ **IMPORTANT!** Live Update has been superseded by Update Services. If you need to use Live Update, you must also use the Legacy Playlist.

Live Update can be used to update data by adding the `OnLiveUpdate` function to the Script Editor. This function is called at regular intervals when the template is on air after being run from the playlist in Director. The function can be used to continuously update the scene with new data (for example stock quotes).

Using the `OnLiveUpdate` function requires that the template is saved (see Save Template) with the Live Update check-box checked.

> ⚠️ **Note:** The update intervals are set in the ContentPilot.ini file.

It's also possible to do live updates using the `OnActionEvent` function, however, this function is not called at regular intervals, only when specific actions are performed in Director (for example Start, Continue etc.).

The Live Update mechanism relies on Director executing the scripts. Director must therefore be up and running at all times when there are Live Update templates in the playlist.

For other control applications such as Viz Trio this is not an optimal workflow. The Update Script mechanism is therefore recommended since it works on the Media Sequencer alone, and is therefore available to all products connected to the Media Sequencer. See Update Script Editor.

> ❌ **Caution:** Having a Director up and running at all times when there are numerous Live Update templates in the playlist will cause Director to load all templates in the background. This will consequently use a lot of memory.

9.14.5 Viz Trio Template

It is possible to create templates that can be used in Viz Trio. The following example and procedure explain how to create a custom user interface for Viz Trio. Note that the Script Example uses specific Viz Trio macro commands. Template Wizard is unable to fully test Viz Trio templates. A Viz Trio client is therefore required.

Script Example

The OnClick event (defined as *Button1Click*) triggers the macro command *TrioCmd* with the parameters to read, set a value, and take page 1000 on air.

```
Sub Button1Click(Sender)
    TrioCmd("page:read 1000")
    TrioCmd("page:set_property 01 ") & yourvalue
    TrioCmd("page:take")
End Sub
```

For more information on macro commands, see the *Viz Trio User Guide*.

Creating a Template for Viz Trio

1. Choose **File** and **New** to create a blank template.
2. Add components and define script events as required (see Script Example).
3. Add Viz Trio macros to the code that will be triggered by the events.

4. Save the template to a file, and copy it to the Viz Trio client machine's **vtwtemplates** directory or shared directory.
5. Start the Viz Trio client and set the template to active in the **Show Settings**.

9.15   Wizard

This section contains the following topics:

- Using the Template Wizard
- Enabling the Prefix and Customize options

9.15.1   Using the Template Wizard

The easiest way to create a template is to use the Wizard (opened from the File menu), and it is also possible to create a template from scratch without adding a scene to it.

1. The first step of the Wizard lets you select a Viz Engine renderer to connect to.

> ⊗ **IMPORTANT!** This step only appears if you have not selected a rendering engine. If no rendering engine is available, you must add one during installation and configured using the Viz Pilot Configuration Tool, see Changing Settings.

> ⚠️ **Note:** The **Launch Viz Preview** button only appears if a preview engine has been installed and configured on the Template Wizard client machine, see Local Preview in Template Wizard.

2. The Wizard's second step is where the template scenes are selected. The folder structure (left) shows what is available on the currently connected Viz Engine. Scenes displayed in the middle pane can be dragged and dropped from the scene display pane onto the Selected Scenes pane (right).

> ✅ **Tip:** If a scene is loaded on Viz, it will not take into account any changes made by the scene designer after it was loaded; hence, it is good practice to clean up or restart Viz Engine before adding scenes for template generation.

- **Objectify All / Objectify Selected:** Generate GEOM objects from all or the selected transition logic front scenes. The GEOM objects are used by the background scene. If transition logic scenes are exported to a different Viz Graphic Hub, and there already exists a template for them, the GEOM objects must be regenerated for the scene to work properly when played out or previewed. This feature does not require the template to be rebuilt.
- **Finish:** Finishes the template creation with a default layout and selection of exposed objects.

3. The Wizard's final step shows a list of exposed and editable objects. The objects are numbered and each object property can be shown and subsequently accessed in the template.

- **Name:** Shows the names of the objects properties.
- **Description:** Shows the descriptions of the objects properties. The text is fetched from the description fields of the corresponding Control plugins in the scene.
- **Type:** Shows the data types of the objects properties.
- **Add:** Sets which properties to include in the template.
- **Prefix charactercount:** Sets the number of prefixes to be used when generating a template with a table style layout. This is used for scenes that do not use control plugins like Control List or Control Chart for table creation. For example, if a scene consists of four groups, and each group has three sub-containers, each with text and images, it may be desirable to auto-generate a table layout representing the sub-groups rather than having a long list of form elements on the template form. Remember to set the Field Identifiers for each group. For example: Group 1 with 011,012,013, and Group 2 with 111, 112, 113 and so on.
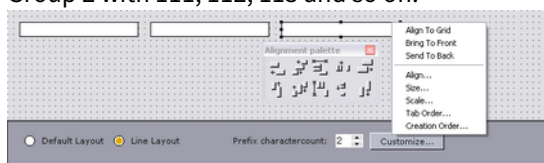
- **Customize:** Arranges the components before the template is generated.

9.15.2   Enabling the Prefix and Customize options

Add the key `linelayout_visible__` to the *ContentPilot.ini* file under the Template Wizard section with the value y (Yes).

9.16 Template Editor

The template editor is used to create and edit a template's user interface. Templates can be created from scratch, or by using the Wizard to select the scene the template is to control. Using the Wizard adds a default set of components to the template during scene import.

This section contains the following topics:

9.16.1 Template Editor Functions

The components added when using the Wizard tool are based on the Exposed Objects information in the scene; however, components can be replaced and new ones added to extend or change the functionality of the template and to change the appearance and behavior of the scene.

Note that the template name (square brackets) and the scene path is visible in the template window title bar. For standard scenes the scene name is used; however, for transition logic scenes the background scene name is used.

> ❌ **IMPORTANT!** Always remember to test templates before deploying them.

Template Editor Context Menu
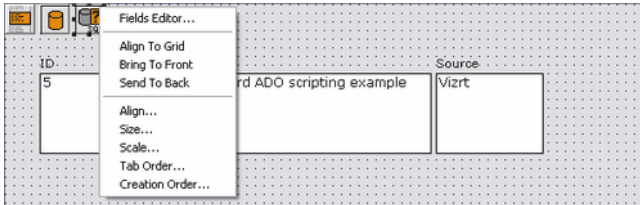
The context menu in the Template Editor contains the following functions:

- **Edit Image**...: Adds a Static Image Component to be used as a template background, for example. Only available when using Static Image Component.

> ❌ **IMPORTANT!** When using the static image component, use compressed image types such as JPG or lossless ones such as PNG to conserve space. The image does not need to be small in terms of pixel size. The whole template is streamed to the database as a BLOB, and the image data will not be altered. Using JPG or PNG will therefore use much less space than BMP, which is not compressed.

- **Fields Editor:** Adds string components to represent database fields using an active database connection. Only available when using specific Database Components.
- **Align to Grid:** Aligns the template's selected component to the template grid.
- **Bring to Front:** Brings the template's selected component to the front of all stacked components.
- **Send to Back:** Brings the template's selected component to the back of all stacked components.
- **Align:** Lets the user set the Alignment of the template's components.
- **Size:** Lets the user set the size of the template's components (see Size and Scale).
- **Scale:** Lets the user set the scale of the template's components (see Size and Scale).
- **Tab Order:** Lets the user set the Tab Order for the template's components.
- **Creation Order:** Lets the user set the Creation Order for the template's components.

9.16.2   Standalone Templates
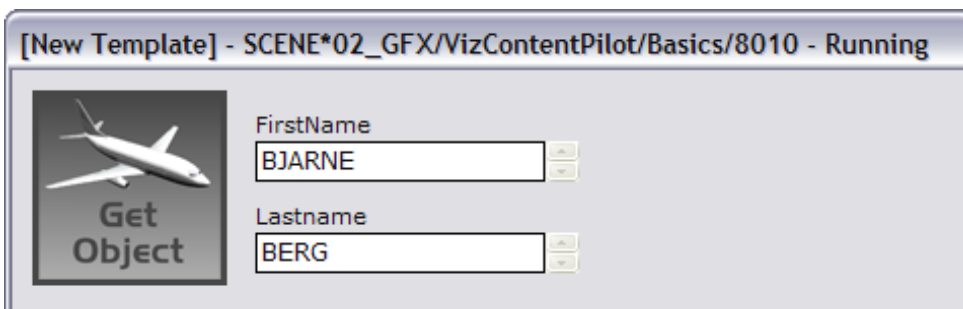
Unlike Transition Logic Templates, standalone templates are only based on a single scene.

Generating a Standalone Template

1. Select **Wizard** on the **File** menu.
2. Add the **scene**.
3. Click **Next**
4. Select the **exposed objects** that are to be added to the template automatically.
5. Click **Finish**.

9.16.3   Transition Logic Templates

Transition Logic Scenes
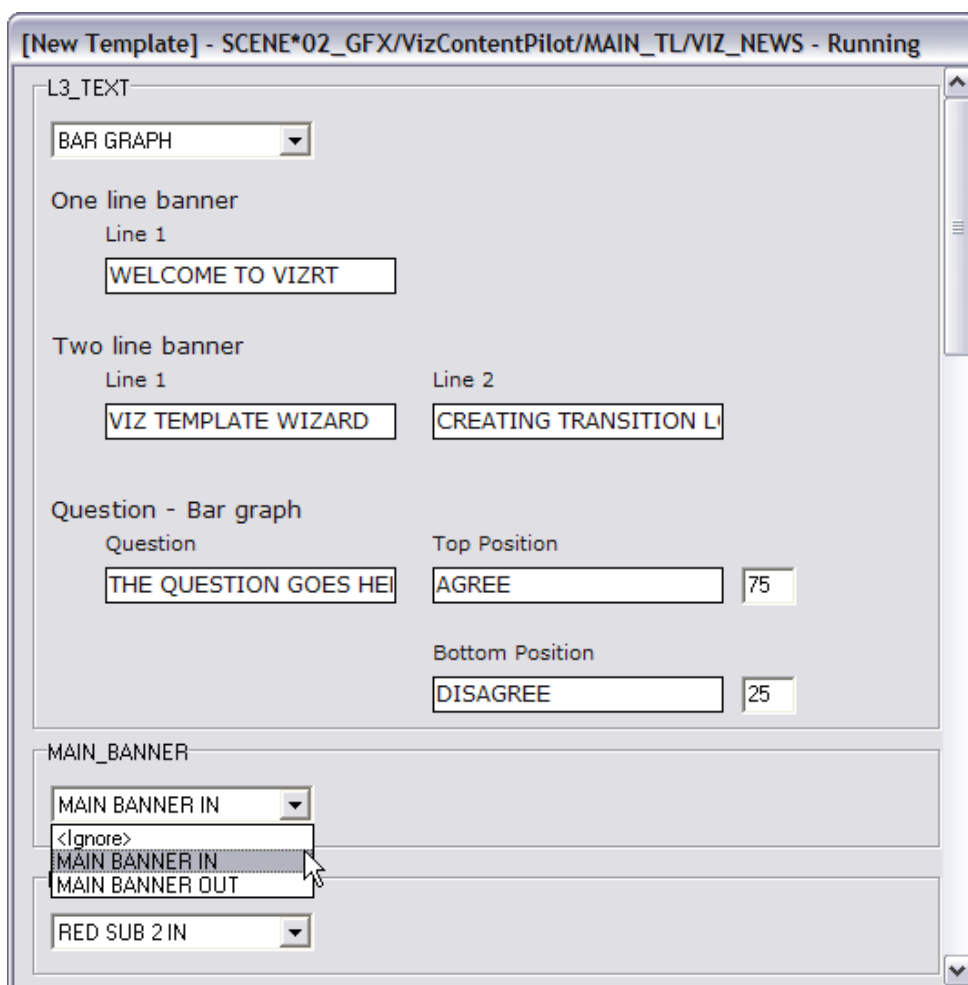
A transition logic scene consists of a background scene and a set of front scenes. When used to create a template in Template Wizard, the front scenes are created as GEOMs (objectified). The objects are added to the background scene in their respective placeholders called layers. Once the template is used to play out the scenes, the front scenes themselves are therefore not used, rather their objects in the background scene are used.

When generating a transition logic template the logical layers, visible in the Layer Panel, are placed on the template in separate group boxes. Each group box also has a drop-down menu with all the GEOMs (the front scenes) for that

layer. The currently selected front scene is displayed in the **ActiveLayerID** property for the **TVTWLayerControl** component.

> ⚠️ **Note:** Although the ItemIndex property is similar to ActiveLayerID, using the ActiveLayerID is recommended as it preserves links to items regardless of changes to the index order or description names in the scene's control object.

A background scene can contain many logical layers. A layer is a placeholder for front scene objects (graphical elements) that are intended to be on air one at a time. In order to have independent elements that can be on-air simultaneously, new logical layers must be added to the background scene.

Each layer can have several states, including everything from position on screen to layout variants or off screen. Certain controls in templates will be visible depending on which layers are currently selected.

Using this method, several independent graphical elements can be on air at the same time and can be taken in and out independently of each other. Transitions can be built into the background in order to obtain a smooth transition between the different states.

Field Identifiers

In order for Template Wizard to generate transition logic templates correctly, one important design convention needs to be followed. As a general rule, all scenes must have *unique* **Field identifiers** for all Control plugins. The identifier can be both text and numbers but must not contain any spaces.

There is one exception where it is permitted and even useful to have the same Field identifier on multiple Control plugins. The same Field identifier for values of the same data type can be used for scenes that use the same logical layer, this means that they will not be on-air at the same time. In the template, these values can be set in a single editor rather than in several editors; one set for each scene, when using unique Field identifiers for all.

The description label for the editor will be the same so it must work logically for both objects, otherwise the user could become confused.

An example of why it can be useful to use the same identifier, is when there are two OTS graphics with an image and a text in the same logical layer, one on the left side and one on the right side, it is possible to name both Control Text plugin Field identifiers *OTS_Text*, and the Control Image plugin Field identifiers *OTS_Image*.
With common identifiers for the two OTS text elements and the two OTS image elements, the Wizard will only generate one set of components. This setup will work well since the elements will never be on screen at the same time.

Generating a Transition Logic Template

1. Select **Wizard** on the **File** menu.
2. Add the **front scenes** that are controlled by the background scene.
3. Click **Next**
4. Select the **exposed objects** that should be added to the template automatically.
5. Click **Finish**.

> ⚠️ **Note:** If a background scene is selected, an error message is shown (The background scene cannot be selected!).

9.16.4   Template Testing

It is good practice to create procedures for testing templates. The procedure below is an example of how to best test templates before using them in a production environment.

Testing Templates

1. Design and test scenes in Viz Artist.
2. Double check all Control Objects in Viz Artist.
3. Using the Wizard tool, design the template and save it in a test Concept.
4. Clean-up the template layout and do a manual and final design.
5. Test the template in Template Wizard.
    - Playout and Control Objects
    - Scripts
    - Special Characters
6. Test the template in Viz Pilot News.
    - Tab order of all components
    - Default save name (& Save Template window on/off if used)
    - Variants
    - *Optional*: Crop Tool
7. Test the playout of data elements from a control room with Director.
8. Deploy the template to journalists.

9.16.5   Aligning

Components can be positioned manually by drag and drop, or by selecting multiple components and using the built-in alignment functions. It's also possible to snap components to the template grid.

Positioning Components
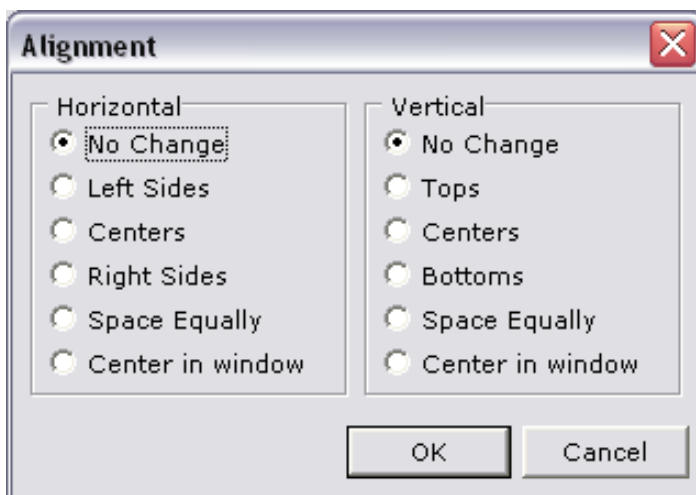
1. Select the components to align.
2. Right-click on the template and select **Align**.

3. Set the horizontal and vertical alignments and click **OK**.

Positioning Components by Snapping To Grid

Position components to the template grid automatically by selecting **Snap to Grid** on the **View** menu.

9.16.6   Size and Scale

Setting the size and scale of components can be done per component or by selecting a group of components, and can be used to make text boxes and other components of the same type with the same size and/or scale.

- **Size: Width/Height:** Sets the width or height of selected components.
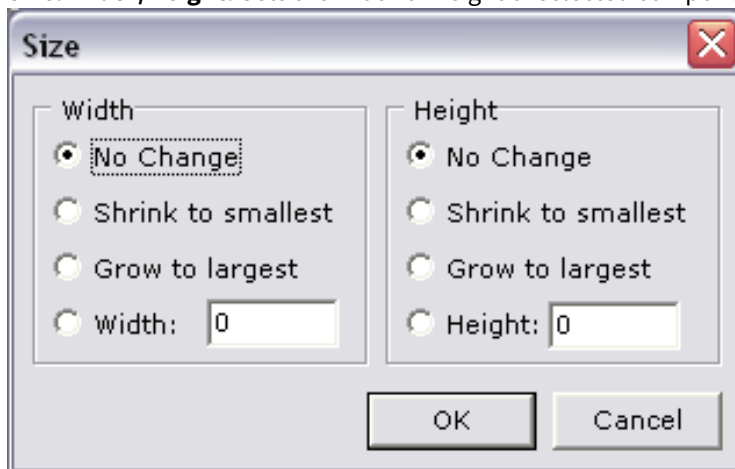
- **No Change:** No changes are applied to the width or height of the selected components.
- **Shrink to smallest:** Shrinks the size of all components to the same as the smallest.
- **Grow to largest:** Grows all components to the same size as the largest component.
- **Width/Height:** Sets a custom width/height for the selected components.
- **Scale: Scaling Factor:** Sets the scaling factor in percentage.
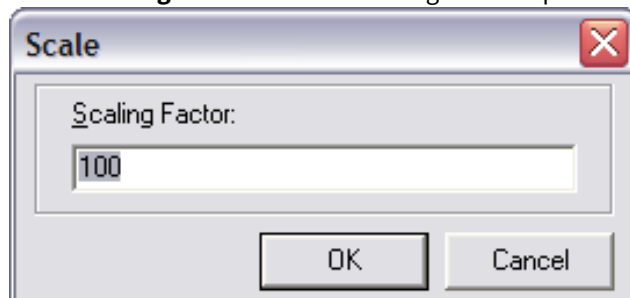
Setting Component Size

1. Select the components to size by pressing **SHIFT** and clicking the components.
2. Right-click the template, and on the context menu select **Size**.
3. Set the **Width** and **Height** properties and click **OK**.

Setting Component Scale

1. Select the component to scale by pressing **SHIFT** and clicking the components.
2. Right-click the template, and on the context menu select **Scale**.
3. Set the **Scaling Factor** (percentage) and click **OK**.

9.16.7   Tab Order

Press the **TAB** key to move between the different fields (components) on the template.

Setting the Tab Order

1. Right-click on the template and on the context menu select **Tab Order...** .
2. Arrange the order by selecting items and using the arrow buttons, or drag and drop, to move them up and down.
3. When done click **OK**.

9.16.8   Creation Order
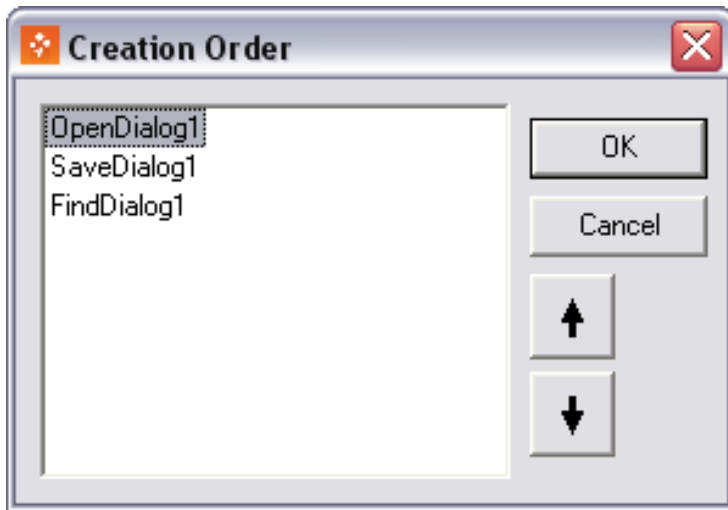
Use the **Creation Order** dialog to specify the order in which the template creates non-visual components. The order is used when the form is loaded at design time or run time. A non-visual component is a component that is visible when designing, but has no appearance at runtime until it is called (for example SaveDialog), or simply never appears at runtime (for example TWComPort).

The List box only displays non-visual components on the active template, their type, and their current creation order. The default creation order is determined by the order in which the non-visual components are placed on the template.

9.16.9   Background Images

The use of large size background images (in bytes) in templates is not recommended. This is made possible by the Static Image Component component; however, for every data element that uses the template, a copy of the template is created and saved on the database, including the background image.

9.16.10   Tags

Tags are administered through the Pilot Data Server Web Interface. Templates can then be tagged by selecting a tag in the Save Template dialog.

Accessing Tag Settings

1. Select the **Pilot Data Server Web Interface** from the Windows Start Menu.
2. Click the **Settings** link.
3. Click the **Tag settings** link.

Defining a New Tag

1. Click the **Add tag** button on the **Tag settings** page.
2. Enter a name for the tag in the New Tag field.
3. Click **OK**.

Renaming a Tag

1. On the **Tag settings** page, click the **Rename** icon next to the desired tag.
2. Enter a new name for the tag in the **Rename Tag** field.
3. Click **OK**.

Deleting a Tag

1. On the **Tag settings** page, click the **Delete** icon next to the desired tag.
2. Click **OK** to confirm the deletion.

> ⚠️ **Note:** Deleting a tag does not delete the templates that refer to the tag.

Assigning a Tag to a New Template

1. Create a new template.
2. Click **Save**.
3. Enter Template name and Concept as required.
4. Select a tag in the **Tag** dropdown.
5. Click **Save as New.**

Assigning or Modifying an Existing Template's Tag

1. Open an existing template.
2. Click **Save**.
3. Select a tag in the **Tag** dropdown.
4. Click **Overwrite existing**.

Removing an Assigned Tag

1. Open an existing template.
2. Click Save.
3. Select <None> in the Tag dropdown.
4. Click **Overwrite existing**.

10   Template Wizard Components

This section contains the Template Wizard's components and their properties:

- Object Inspector
- Additional Components
- Communication Components
- Dialog Components
- Viz Control Components
- Other Components
- Component Properties
- Import Components
- Standard Components
- Database Components

⚠️ **Note:** You can use the Component Palette to customize which components appear in the dropdown list at the top of Template Wizard.

10.1   Object Inspector
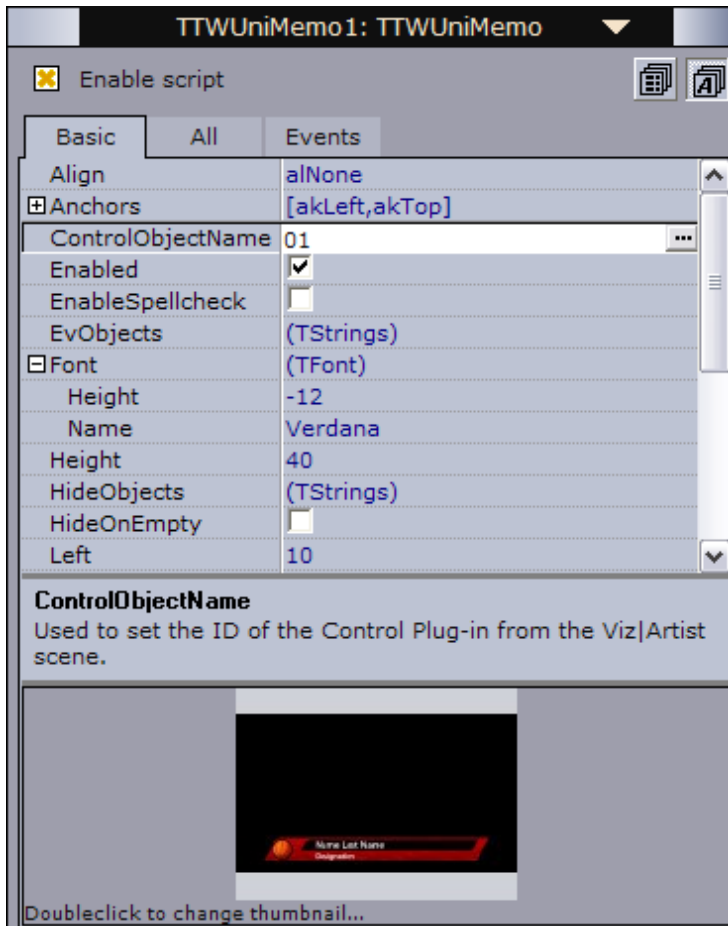
Object Inspector displays Component Properties which can then be configured, and Component Events can be added through scripting. All properties use the ControlObjectName property that ties the Exposed Objects or Scene Objects (see the Resource Panel) from the graphics scene to the template.

The majority of components contain many properties, most of which are only used in special cases. The inspector therefore has a separate properties views.

It is also possible to sort properties by category or name by clicking the sort buttons. Properties are sorted by name by default.

It is also possible to use the inspector to change the Template Thumbnail.

> ⚠ **Note:** The property editor in Template Wizard does not support Unicode directly. To edit properties like "UTF8Text", press the ellipsis button (**...**) and then edit the text.

10.1.1   ControlObjectName
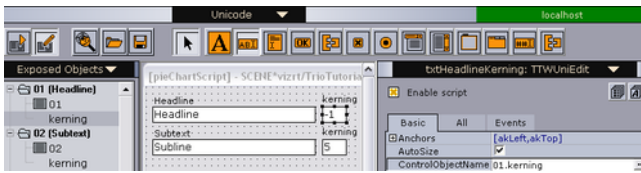
All components that support graphic scenes with Control plugins contain the ControlObjectName _property which shows the exposed object to which a property is connected. The value is linked to the structure and numbering of the exposed objects. The value below the **_01 (Headline)** entry named **01** contains the default value since its name is the same as the main entry - **01(Headline)**. This is a text string in the example above. The ControlObjectName for this value is then 01. The kerning value for the text string is grouped under the 01 node and is given the ControlObjectName **01.kerning**. The naming convention is *objectname.propertyname*.

> ⚠ **Note:** It is good practice to name the scene's Field Identifiers 01, 02, and 03 instead of 1,2,3,4 ... 11,12 and so on. Using 1,2,3,4 ... 11,12 will cause data to be assigned to the wrong fields when using Import Components since the Media Sequencer will place 11 and 12 before 2 and so on.

To See a Component's Properties

Start Template Wizard, add a component to the template, and use the Object Inspector on the right to browse through the different properties.

10.1.2   Component Events

If scripting is enabled for a template (see Template Wizard Components), each component will show the Events tab. The Events tab lists all possible events that can call scripts when certain user actions, functional or procedural events are triggered.

Example 1

Scenario: A user clicks a button on a template and an `OnClick` event triggers a new event that loads a piece of data into a dialog box.
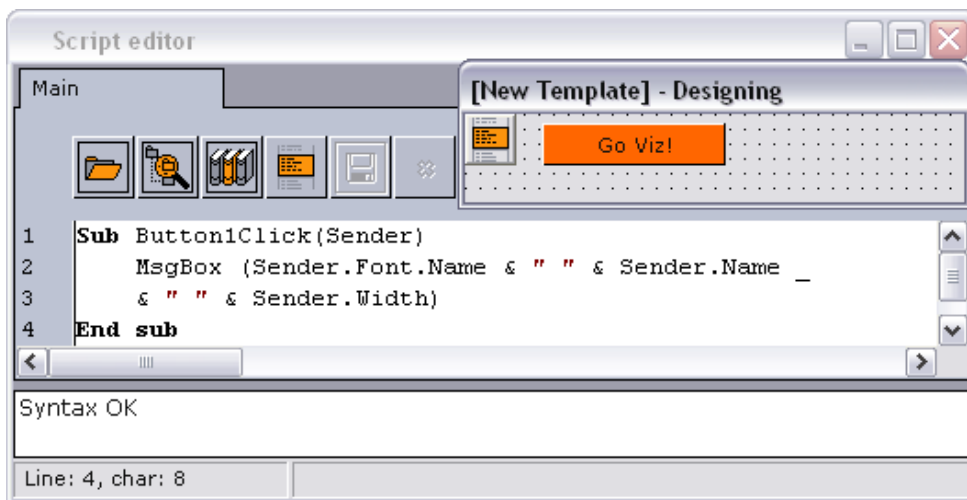
The image above shows a solution to the scenario with an **OnClick Event** where the button's `Sender` object is an argument to the Sub procedure `Button1Click` . The event name is comprised of the Name (`Button1`) property and the event type (`OnClick`). The `Sender` object contains all the component properties of the sender.

> ⚠️ **Note:** Do not confuse the *Name* property with the *Caption* property. The *Caption* property is what is displayed on the button, and the *Name* property is what is used to reference the button.
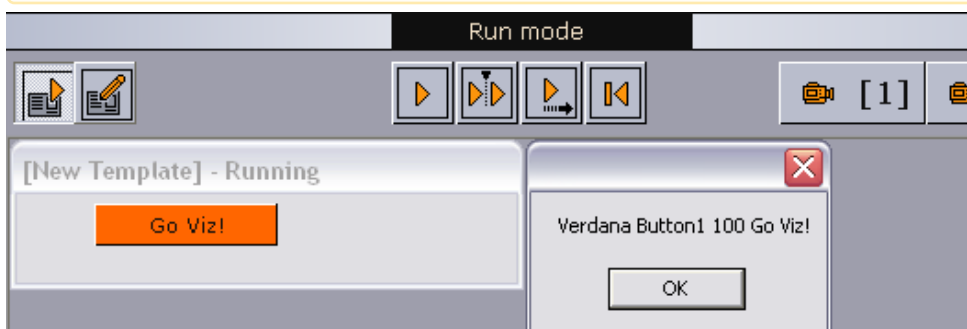
Clicking the button triggers the `OnClick` event, and displays a message box with the button's font type, name, width in pixels and its label.

Example 2

Scenario: A user adds two numbers to summarize a score result.

The scenario from example 1 above can be extended to fit this example by using the same event on several components, and thereby linking them to the same piece of code. The example below uses a small piece of code with one global value, two procedures and a function to calculate the value.

```
Dim globalTotalValue
Sub clearClick(Sender)
    globalTotalValue = 0
    txtValue1.Text = ""
```

```
        txtValue2.Text = ""
        txtSum.Text = ""
    End sub
    Sub addValue(Sender)
        calculate(CInt(Sender.Text))
    End sub
    Function calculate(value)
        globalTotalValue = globalTotalValue + Value
        txtSum.Text = globalTotalValue
    End Function
```

Calculate

Value 1 `10`
Value 2 `20`
Total `30` Clear

The two text fields are named Value 1 and Value 2, and both have the `OnExit` event defined as `addValue` . Entering a value in field 1 and exiting the field will trigger the event `addValue` which sends the value (`Sender.Text`) to the calculate function. The calculate function calculates the value and sets the new `globalTotalValue` in the text field, named `txtSum` . The two value fields share this small piece of code.

Adding an Event

1. Check the **Enable script** option in the Object Inspector (**CTRL + F12**) to enable scripting.
2. Click the **Events** tab, and double-click the empty field to the right of the event. This creates a default event name and inserts a script framework (sub procedure) into the script editor.

10.1.3   Template Thumbnail

Doubleclick to change thumbnail...

All templates use the thumbnail generated when the scene was saved in Viz Artist. Multi-scene templates (for example Transition Logic) use the thumbnail from the first scene that is added to the Wizard's Selected scenes pane (Wizard's step 2).

It is also possible to select a custom image (such as jpeg, bmp) by double-clicking the Thumbnail image (lower-right) in Template Wizard; however, these cannot be based on Viz Artist scene icons.

> ⚠️ **Note:** Thumbnails are visible in Director and Viz Pilot News.

10.1.4   Scripting Support

Visual Basic Scripting (VBScript) is supported by Template Wizard. The supported version is dependent on the installed version on the computer. To find the VBScript version that is installed, locate and view the properties of the *vbscript.dll* file.

> ℹ️ **Example:** C:\WINDOWS\system32\vbscript.dll

For a user and language reference guide on VBScript, see the Microsoft Developer Network (MSDN).

As part of the component descriptions, some VBScript examples are provided to show how the components can be used with scripting.

Enabling Scripting

Check **Enable script** (**CTRL + F12**) to enable scripting.

> ⚠️ **Note: CTRL + F12** toggles between the Template Editor and Script Editor.

See Also

- Resource Panel

10.2   Additional Components

Additional components are components with more specialized properties and features. Additional components are not used as much as standard components.

This section contains the following topics:

- Wrap Fixed Unicode Memo Component
- TrackBar Component
- Mask Edit Component
- Speed Button Component
- DateTime Picker Component
- Timer Component

10.2.1   Wrap Fixed Unicode Memo Component

This component is much like the Unicode Memo. However, there is a significant difference between the two. As the Unicode Memo stores its text as a list of strings, the Wrap Fixed Unicode Memo saves the text in the UTF8Text string property as one long text string. This avoids the problem of line breaks being added to the text provided that *WordWrap* is disabled. Line breaks added by the user are not removed.

10.2.2   TrackBar Component

The TrackBar represents a position along a continuum using a slider and, optionally, tick marks. A track bar can also display a selected range and set minimum and maximum values. The selected range is marked by triangular ticks at the starting and ending positions of the selection.

Notable Properties

- **Max:** Specifies the maximum size of an interval.
- **Min:** Specifies the minimum size of an interval.
- **SelEnd:** Specifies the position of the end point of the selection range.
- **SelStart:** Specifies the position of the starting point of the selection range.
- **Position:** Contains the current position of the slider.

10.2.3   Mask Edit Component

Allows a user to enter and edit data, similar to an edit component, but provides a means to specify particular formats, such as a postal code or phone number. Mask Edit shares the properties of other Text controls.

Notable Properties

- **EditMask:** Specifies the characters a user can enter into the masked edit control to valid characters and formats.
- **Text:** Contains the formatted text based on the mask set by the EditMask property.

10.2.4   Speed Button Component

Use SpeedButton to add a button to a group of buttons in a form. SpeedButton introduces properties that can be used to set graphical images that represent the different button states (selected, unselected, disabled and so on). Use other properties to specify multiple images or to rearrange the images and text on the button.
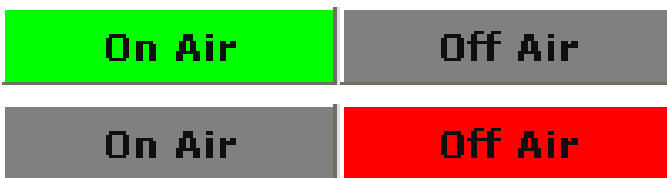
SpeedButton also introduces properties that allow speed buttons to work together as a group. Speed buttons are commonly grouped in panels to create specialized toolbars and tool palettes.

Notable Properties

GroupIndex: Buttons with the same property value (other than 0), work together as a group and can present mutually exclusive choices to the user.

Script Example

```
Sub InitForm
    SpeedButton1.Color = clGray
    SpeedButton2.Color = clRed
End Sub
Sub SpeedButton1Click(Sender)
    SpeedButton1.Color = clLime
    SpeedButton2.Color = clGray
End sub
Sub SpeedButton2Click(Sender)
    SpeedButton1.Color = clGray
    SpeedButton2.Color = clRed
End sub
```

10.2.5   DateTime Picker Component

The DateTime Picker component is used for entering dates or times. In dmComboBox date mode, DateTime Picker resembles a list box or combo box, except that the drop-down list is replaced with a calendar illustration. Users can select a date from the calendar. Dates or times can also be selected by scrolling with the Up and Down arrows and by entering text. Use the `.Date` and `.Time` property to access the selected date/time from script.

Notable Properties

- **Date:** Indicates the date entered by the user.
- **DateFormat:** Determines whether the date format is long or short.
  DateMode: Can be used with a drop-down calendar or by the Up/Down arrows with which the user can adjust the date (only applied with dtkDate).
- **Kind:** Determines if date or time is shown.
- **Time:** Indicates the time entered by the user.

10.2.6 Timer Component

Use the Timer component to trigger an event after a measured interval. Enter the code that is to occur at the specified time inside the timer component's OnTimer event.

Notable Properties

- **Interval:** Determines how frequently the OnTimer event occurs. Specified in milliseconds. Default is 1000 (one second)
- **Enabled:** Determines if the timer is active or not. The OnTimer event only occurs when the Enabled property is True.

Script Example

The Timer event gets called at the interval specified (1000 = 1 second) and the current date and time is set in a text box.

When the template is loaded it will be in Start mode.

```
Sub Initform
    Timer1.Enabled = False
    TWColorButton1.Color = clLime
    TWColorButton1.Caption = "Start"
End Sub
```

When started, the Timer1Timer event adds the time **Now** to the combo box.

```
Sub Timer1Timer(Sender)
    TWUniComboBox1.Items.Add(Now)
End sub
```
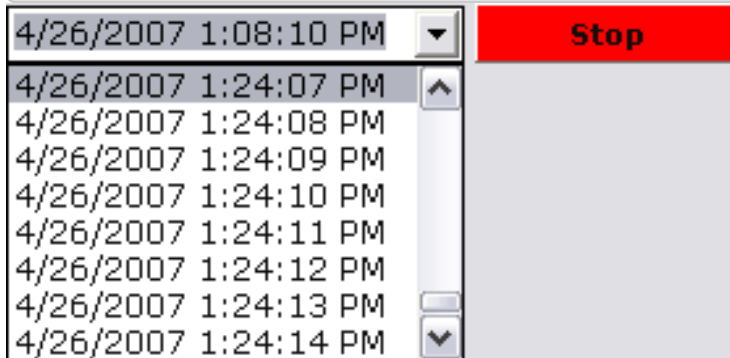

When the button is clicked, it either sets Timer1 to true or false, and at the same time changes color and caption.

```
Sub TWColorButton1Click(Sender)
    if (Timer1.Enabled = True) Then
        Timer1.Enabled = False
        TWColorButton1.Color = clLime
        TWColorButton1.Caption = "Start"
    Else
        Timer1.Enabled = True
        TWColorButton1.Color = clRed
        TWColorButton1.Caption = "Stop"
    End If
End sub
```

10.3   Communication Components

Use the Communication components to enable communication with other computers.

This section contains the following topics:

- Com Port Component
- Generic Com Port Component
- Client Socket Component

10.3.1   Com Port Component

The TTWComPort component is used to send and receive data through a com port.

Notable Properties

- **Baud:** Sets the baud rate, for example at 19200.
- **ComNumber:** Sets the com port number.

Script Example

The TTWComPort has the following methods for sending data:

- **.PutString("Hello"):** sends the string "Hello"
- **.PutChar(Asc("A")):** sends the character "A"
  To receive data, data triggers must be registered on the TTWComPort. The data trigger will be fired whenever the incoming data matches a given trigger.

> ✅ **Tip:** A protocol message can start with "START" and end with "END". Register "START" and "END" as triggers so that the user is notified whenever the incoming data matches either "START" or "END".

To register a data trigger, call the AddDataTrigger function of the TTWComPort using the method below:

AddDataTrigger(trigger_text, case_sensitive): The *trigger_text* is the text to match and *case_sensitive* is a boolean value. The result value of the function is a trigger handle that should be stored. In this example, it is named as follows below:

```
SOHDataTrigger = TWComPort1.AddDataTrigger(SOHText, false)
EOHDataTrigger = TWComPort1.AddDataTrigger(EOHText, false)OnTriggerData(CP,
TriggerHandle): Is called whenever the incoming data matches one of the triggers. The
TriggerHandle contains the handle of the trigger that matched the trigger data. It
can be tested like this:
```

```
if TriggerHandle = SOHDataTrigger then
    msgbox "SOH received"
end if
if TriggerHandle = EOHDataTrigger then
    msgbox "EOH received"
end if
```

The OnTriggerData only tests that the incoming data matches one of the registered triggers. To receive the actual data, use the event called OnTriggerAvail. This event is called whenever there is new incoming data. The event looks like this:

OnTriggerAvail(CP, Count), where count is the number of new characters received.

Implement the following function to gather all the incoming data:

```
Sub TWComPort1TriggerAvail(CP, Count)
\q---- Data is available, read it into the sMessage buffer----
    for i=0 to Count-1
        sMessage = sMessage & Chr(TWComPort1.GetChar)
    next
End sub
```

10.3.2   Generic Com Port Component

TTWGenericComPort is a specialized version of the TTWComPort and has a predefined start and end of message characters, 0x01 and 0x04 respectively, and also a start of transmission character 0x02.

Events

It also has an event called OnDataAvailable that is triggered whenever a new message is received. The event looks like this:

- **OnDataAvailable(Sender, MessageHeader, StringList)**
    - **Sender:** Is the TTWGenericComPort component
    - **MessageHeader:** Is the data between 0x01 and 0x02.
    - **StringList:** Is a TstringList with the data between 0x02 and 0x04. Each field in the data is separated with a pipe character '|'.
      A message looking like this:

```
0x01 'My header' 'Field1|Field2|Field3' 0x04
```

causes the OnDataAvailable event to be triggered and the MessageHeader would contain `\qMy header';` the StringList would contain the strings:

```
Field1
Field2
Field3
```

The StringList items can be accessed like this:

```
for i:=0 to StringList.Count-1
    msgbox StringList.Strings(i)
next
```

10.3.3   Client Socket Component

The TClientSocket component is used to connect to a TCP/IP server. It manages the connection when it is open, and terminates the connection when the template is through.

Notable Properties

- **Host:** Host name or IP address of the server to connect to.
- **Port:** Port used to connect to the server.
- **Active:** Indicates whether the socket connection is open and available for communication with other machines.

Basic Methods

- **Open:** Call Open to initiate the socket connection. Open sets the Active property to True.
- **Close:** Call Close to shut down the socket connection. Close sets the Active property to False.
- **Socket.SendText(<Text>):** Use Socket.SendText to write a string to the socket connection.

Script Example

The example below shows how to use the TClientSocket component to send a few commands to Viz Engine.

Trying to connect to the host specified:

```
Sub cbConnectClick(Sender)
    ClientSocket1.Active = False
    ClientSocket1.Host = txtHost.Text
    ClientSocket1.Active = True
End sub
```

A valid connection is established. Enable the send button:

```
Sub ClientSocket1Connect(Sender, Socket)
    cbSend.Enabled = True
End sub
```

10.4　Dialog Components

The following section describes standard Windows dialog boxes such as opening, saving and printing files.

To use the dialog components, they must be linked to a button or another function with VBscript. Some scripting examples follow below.

This section contains the following topics:

- Open and Save Components
- Font, Color, Print, PrintDisplay, Find and Replace Components

10.4.1　Open and Save Components

Open displays a file-selection dialog. Save displays a **Save As** dialog for saving files.

Notable Properties

- **DefaultExt:** Specifies a default file extension.
- **FileName:** Indicates the name and directory path of the last file selected.
- **InitialDir:** Determines the current directory when the dialog opens.

Script Example

The example below shows an Open and Save dialog that opens a file reading the content into a multiline edit box. The template uses the OpenDialog and SaveDialog components. Note the settings of the Filter and Title properties in the Object Inspector. To invoke a dialog, call its *.Execute* method. Execute opens the file-selection dialog, returning True when the user selects a file and clicks **Open** or **Save**. If the user clicks **Cancel**, Execute returns False.

```
const ForReading = 1
const ForWriting = 2
const ForAppending = 8
\q OpenDialog example
Sub TWUniButton1Click(Sender)
    if OpenDialog1.Execute then
        TWUniMemo1.Lines.LoadFromFile(OpenDialog1.Filename)
    end if
End sub
\q SaveDialog example
Sub TWUniButton2Click(Sender)
    if SaveDialog1.Execute then
        TWUniMemo1.Lines.SaveToFile(SaveDialog1.Filename)
    end if
End sub
```

The example below shows an Open dialog using the File System Object to read the content of the chosen file. Documentation about the File System Object can be found at: http://www.microsoft.com/scripting:

```
Sub TWUniButton3Click(Sender)
if OpenDialog1.Execute then
    Set fso = CreateObject("Scripting.FileSystemObject")
    Set ts = fso.OpenTextFile(OpenDialog1.Filename, ForReading)
\q The file has been opened, now read the content
    TWUniMemo1.Text = ts.ReadAll
\q Close the file
    ts.close
\q Clean up
    set ts = nothing
    set fso = nothing
end if
End sub
```

10.4.2 Font, Color, Print, PrintDisplay, Find and Replace Components

The Font, Color, Print, PrintDisplay, Find and Replace dialogs all display generic Windows dialogs for performing operations specific to their properties and methods.

Notable Properties

- **Color:** Specifies the background color of the control.
- **Font.Name:** Identifies the typeface of the font.

Script Example

The following script uses one button and executes the Font and Color dialog:

```
Sub TWUniButton1Click(Sender)
    FontDialog1.Execute
    TWUniButton1.Font.Name = FontDialog1.Font.Name
    ColorDialog1.Execute
    TWUniButton1.Color = ColorDialog1.Color
End sub
```

As with the Open and Save dialogs, the Execute method is used above. This small script lets you select and set the font and color for the button component.

10.5   Viz Control Components

This section covers the following topics:

- Drag Edit Component
- Value Control Component
- Audio Control Component
- Targa Control Component
- Triplet Editor Component

10.5.1   Drag Edit Component

The TDragEdit component is used to set numeric values, either by entering the values using a numeric pad or using the cursor by clicking the text box and dragging in a horizontal direction to adjust the value.

Notable Properties

- **Enabled:** Enables the control.
- **ReadOnly:** Enables the value to be read only.
- **Text:** Sets the text string associated with the control.

10.5.2   Value Control Component

The TWValueControl component is a generic component used for scripting to control values outside the scope of Template Wizard's default components. It is therefore a good tool to use in situations where Template Wizard does not offer a default component for certain exposed Control plugins and properties.

Value Control can only be used with scripting, and is therefore not visible as a component to the operator. However, you can expose it by using other components that can be used to assign values to the Value Control's `ControlValue` property.

Notable Properties

- **ControlObjectName:** Sets the exposed Control plugin property that Value Control is to control (for example 01.kerning).
- **ControlValue:** Sets the value that Control Object in Viz is to receive in order to control the exposed Control plugin property.

Script Example

```
Sub TWUniButton1Click(Sender)
    TWValueControl1.ControlValue = TWUniEdit1.Text
End sub
```

10.5.3   Audio Control Component

The VizAudioCtrl component is used to select audio clips from a defined directory (for example share). Supported file formats for this control are mp3 and wav.

Notable Properties

- **_InitialDir:** Sets the default directory/share from where the audio clips are to be located.
- **ChannelID:** Sets the Channel ID used by the scene in Viz for playing out audio.
- **Director:** Sets the name of the director in the scene that controls the audio animation.
- **Filename:** Sets the audio filename.
- **Keyframe:** Specifies the Viz keyframe to control.

> ⚠ **Note:** The `_UnixShare` property is deprecated.

10.5.4   Targa Control Component

The VizTargaCtrl component is used to selected video clips from a defined directory (for example share). Supported file formats for this control are dif, avi and dv.

Notable Properties

- **_InitialDir:** Sets the default directory/share from where the video clips are to be located.
- **ChannelID:** Sets the Channel ID used by the scene in Viz for playing out video.
- **Director:** Sets the name of the director in the scene that controls the video animation.
- **Filename:** Sets the video filename.
- **Keyframe:** Specifies the Viz keyframe to control.

> ⚠ **Note:** The `_UnixShare` property is deprecated.

10.5.5   Triplet Editor Component

The TripletEditor component is used to set control values exposed from Viz such as position and scaling that require control of three values (for example X, Y and Z).

Notable Properties

- **Max, Min:** Sets the maximum and minimum values.
- **XValue, YValue, ZValue:** Sets the parameters X, Y and Z.

10.6   Other Components

This section describes components that are not part of the default setup, or are based on specialized Viz plugins used by designers to enable operator control over elements in the scene:

- Table Editor Component

10.6.1   Table Editor Component

The TTWTableEditor component is derived from the Control List plugin; hence, it is only possible to use if the Control List plugin is part of a scene. The Table Editor component allows the Table Editor to be used when the scene is edited in Viz Pilot News or Director.

Notable Properties

- **ControlObjectName:** Refers to the *Field Identifier* set for the Control List plugin.
- **Name:** Sets the name of the table object.

Functions

- **GetTableController:** Gets the table controller which can be assigned to a variable.

GetTableController Functions and Procedures

Name	Description
setCellValue	*procedure setCellValue(rowIdx: Integer; colIdx: Integer; const value: WideString);* **rowIdx**: the 0 based row index to use. **colIdx**: the 0 based column index to use. **value**: The new Cell value.
getCellValue	*function getCellValue(rowIdx: Integer; colIdx: Integer): WideString;* **rowIdx**: The 0 based row index to use. **colIdx**: The 0 based column index to use. Returns the current cell value at the supplied index.
getRowCount	*function getRowCount: SYSINT;* Returns the number of rows

Name	Description
getColumnCount	*function getColumnCount: SYSINT*; Returns the number of columns.
insertRow	*procedure insertRow(Idx: Integer);* **Idx**: The 0 based row index to use. Inserts an empty row at index Idx.
deleteRow	*procedure deleteRow(Idx: Integer);* **Idx**:The 0 based row index to use. Deletes the row at index Idx.
moveRow	*procedure moveRow(OldIdx: Integer; NewIdx: Integer);* **OldIdx**: The 0 based row index to move from. **NewIdx**: The 0 based row index to move to. Moves a row from old Idx to NewIdx

Script Example

If the name of the Table Editor component is **Table**, the table controller can be used to assign the table object to a variable. The GetTableController functions and procedures can be used to manipulate the table.

```
Sub SetValueClick(Sender)
    Dim TableController
    Set TableController = GetTableController(Table)
    TableController.setCellValue 0,1,TWUniEdit1.Text
End Sub
```

10.7   Component Properties

10.7.1   Component Properties Listed in the Object Inspector

Action	Designates the action associated with the control.
Active	Use Active to determine or set whether a data set is populated with data. When Active is false, the data set is closed; the data set cannot read or write data and data-aware controls cannot use it to fetch data or post edits. When Active is true, the data set can be populated with data. It can read data from a database or other source (such as a provider).
ActivePage	Specifies the page currently displayed by the page control.
AliasName (Database)	Specifies the Borland Database Engine (BDE) alias used by this database connection
Align	Determines how the control aligns within its container (parent control).
Alignment	Controls the horizontal placement of the text within the label.
AllowEditingOf DefaultName	Determines if the user is allowed to edit the suggested name in the save dialog.
AllowGrayed	Determines whether check box can be in a grayed state.
AllowUp	Specifies whether all speed buttons in the group that contain this speed button can be unselected at the same time.
Anchors	Specifies how the control is anchored to its parent. Use Anchors to ensure that a control maintains its current position relative to an edge of its parent, even if the parent is resized. When its parent is resized, the control holds its position relative to the edges to which it is anchored. If a control is anchored to opposite edges of its parent, the control stretches when its parent is resized.
Attributes	Set Attributes to specify whether a connection object performs retaining commits or retaining aborts. The two behaviors are independent of each other. Attributes can contain one of the two TXactAttribute values (xaCommitRetaining and xaAbortRetaining), both values, or neither value.

AutoCalcFields	Determines when the OnCalcFields event is triggered. Set AutoCalcFields to control when the OnCalcFields event is triggered to update calculated fields during data set processing. A calculated field is one that derives its value from the values of one or more fields in the active record, sometimes with additional processing.
AutoCalcFields (Table)	Determines when the OnCalcFields event is triggered.
AutoCloseUp	When AutoCloseUp is true, the drop-down closes up automatically when the user selects an item.
AutoComplete	When AutoComplete is true, the combo box responds to user keystrokes by searching the Items property array for the first item that matches the string entered so far. When AutoComplete is false, this feature is disabled.
AutoDropDown	When AutoDropDown is true, the combo box automatically drops down its list when the user starts typing a string while the combo box has focus. When AutoDropDown is false, the user must explicitly use the drop-down button to drop down the combo box list.
AutoEdit	Determines if a data source component automatically calls a data set's Edit method when a data-aware control associated with the data source receives focus.
AutoRefresh	Specifies whether server-generated field values are automatically re-fetched.
AutoScroll	Use AutoScroll to specify whether the page scroller scrolls when the mouse hovers over one of its arrows.
AutoSelect	Determines whether all the text in the edit control is automatically selected when the control gets focus.
AutoSize	Determines whether the size of the label automatically resizes to accommodate the text.
BevelEdges	Use BevelEdges to get or set which edges of the control are beveled. The BevelInner and BevelOuter properties determine the appearance of the specified edges.
BevelInner	Determines the style of the inner bevel of a panel.
BevelKind	Use BevelKind to modify the appearance of a bevel. BevelKind influences how sharply the bevel stands out.
BevelOuter	Determines the style of the outer bevel of a panel.

BevelWidth	Determines the width, in pixels, of both the inner and outer bevels of a panel.
BiDiMode	Specifies the bi-directional mode for the control. Use BiDiMode to enable the control to adjust its appearance and behavior automatically when the application runs in a locale that reads from right to left instead of left to right.
BorderStyle	Determines whether the edit control has a single line border around the client area.
BorderWidth	Use BorderWidth to specify how wide the border around the panel should be. A value of 0 (zero) means no border will appear.
CacheSize	Set CacheSize to control how many rows the ADO data set's provider keeps cached in its buffer and how many to retrieve at one time into local memory. Default value of CacheSize is 1 and the minimum allowed value is 1.
CachedUpdates	Specifies whether cached updates are enabled for a data set.
CalAlignment	Set CalAlignment to dtaLeft or dtaRight to align the drop-down calendar on the left or right side of the component. This property is applicable only if the Kind property is dtkDate and the DateMode property is dmComboBox.
CalColors	Use the CalColors property to get or set the calendar's background color, month background color, text color, title background color, title text color, and trailing text color.
Cancel	Determines whether the button's OnClick event handler executes when the Escape key is pressed.
Caption	Specifies a text string that identifies the control to the user.
Center	When the image does not fit perfectly within the image control, use Center to specify how the image is positioned. When Center is true, the image is centered in the control. When Center is false, the upper left corner of the image is positioned at the upper left corner of the control. The default value for Center is false.
ChannelID	Specifies the Viz Engine channel that should be used.
CharCase	Determines the case of the text within the edit control.
Checked	Specifies whether the check box appears checked.
Cmd	Deprecated (no longer in use).

Collate	Collate is true whenever the Collate check box is selected in the Print dialog. To make the dialog open with the check box selected, set Collate to true in the Object Inspector or in program code.
Color	Specifies the background color of the control.
Columns	Use Columns to specify the number of columns, in a multi-column list box, that are visible without having to use the horizontal scroll bar.
CommandTime Out	Use CommandTimeout to specify the amount of time, in seconds, that can expire before an attempt to execute a command is considered unsuccessful. The default value is 30 seconds.
Connected	Set Connected to true to establish a connection to an ADO data store without opening a data set. Set Connected to false to close a connection. The default value for Connected is false.
Connection	Use Connection to specify an ADO connection object to use to connect to an ADO data store.
ConnectionStrin g	Set ConnectionString to specify the information needed to connect the ADO connection component to the data store. The value used for ConnectionString consists of one or more arguments ADO uses to establish the connection. Specify multiple arguments as a list with individual arguments separated by semicolons.
ConnectionTim eOut	Use ConnectionTimeout to specify the amount of time, in seconds, that can expire before an attempt to make a connection is considered unsuccessful. The default value is 15 seconds.
ConnectOptions	Set ConnectOptions to specify whether the connection established by the TADOConnection is synchronous or asynchronous. The default value for ConnectOptions is coConnectUnspecified.
Constrained	Indicates whether updates and inserts, that don not conform to the result set generated by a SELECT statement, are permitted for Paradox and dBASE tables.
Constraints	Specifies the size constraints for the control.
Constraints (DataSet)	Specifies record-level constraints that must be met when editing the data.
ControlObjectN ame	Used to map the Field Identifier ID set for the scene. See also ControlObjectName and Control plugins.

ControlValue	The value that the control object in Viz Engine should receive.
Copies	If Copies is 0 or 1, the dialog will have 1 in the Number of Copies field. To make the dialog open with 2 or more copies selected, set the value of Copies in the Object Inspector or in program code.
Ctl3D	Determines whether a control has a three-dimensional (3-D) or two-dimensional look.
Cursor	Specifies the image used to represent the mouse pointer when it passes into the region covered by the control
CursorLocation	Use CursorLocation to indicate whether the cursors that use the connection object to connect to the ADO datastore use a client-side or server-side cursor library. CursorLocation only affects connections opened after the property is set. The default value for CursorLocation is clUseClient.
CursorType	Set CursorType to indicate the type of cursor the ADO data set uses for the record set when it is opened. CursorType must be set prior to activating the data set component. Among other cursor aspects, CursorType affects directional scrolling through a record set and the visibility of changes made by other users. The default value of CursorType is ctKeyset.
CustomColors	Each custom color is represented as a string of the form ColorX=HexValue. For example, the following string sets the first custom color
DatabaseName	Specifies the name of the database associated with this data set.
DataField	Specifies the field from which the edit control displays data.
DataSet	Specifies the data set for which the data source component serves as a conduit to data-aware controls. Set DataSet to the name of an existing data set component either at design time, or at runtime. By changing the value of DataSet at runtime an application can effectively use the same data-aware controls to display and edit data in different data sets.
DataSetField	Use DataSetField when the data set is the nested detail set of another data set, such as an Oracle8 nested data set. Read DataSetField to access the data set field component in the master data set that includes this data set as a detail. Set DataSetField to a field component in the master data set to assign this data set as the value of that field.
DataSource	Links the edit control to the data set that contains the field it represents.
DataSource	Specifies the data source component from which to extract current field values to use with same-name parameters in the query's SQL statement.

Date	Use Date to get or set the date that is marked on the calendar. The value of Date must lie within the range specified by the MaxDate and MinDate properties. If MultiSelect is true, the selected range of dates goes from Date to EndDate.
DateFormat	If DateFormat is set to dfShort, the date is formatted using the short date format specified in the Windows control panel. It appears in this format by default.
DateMode	If DateMode is dmComboBox, the component has a drop-down calendar illustration from which the user can select a date. If DateMode is dmUpDown, the component has Up/Down arrows with which the user can adjust the date. (Applies only when Kind is dtkDate.)
Default	Determines whether the component's OnClick event handler executes when the Enter key is pressed.
DefaultDatabase	Set DefaultDatabase to indicate the database the ADO connection objects uses if the connection to the database specified in ConnectionString is unavailable, the connection cannot be made, or the database is not specified in the connection string.
DefaultExt	DefaultExt specifies a file extension that is appended automatically to the selected file name, unless the selected file name already includes a registered extension. If the user selects a file name with an extension that is unregistered, DefaultExt is appended to the unregistered extension.
DefaultIndex (Table)	Specifies if the data in the table should be ordered on a default index when opened.
DestHeight	The destination height of the image in Viz Engine. If set to <> 0, a crop tool will appear if the selected image does not adhere to the aspect given by DestHeight and DestWidth.
DestWidth	The destination width of the image in Viz Engine. If set to <> 0, a crop tool will appear if the selected image does not adhere to the aspect given by DestHeight and DestWidth.
Device	Use Device to specify whether the dialog should list screen fonts (fdScreen), printer fonts (fdPrinter), or both screen and printer fonts (fdBoth).
DisallowedKeyWords	Defines the disallowed keywords used when searching for media content. When a user searches and finds an image, they are not allowed to select the image if it contains a keyword that is present in the DisallowedKeywords property.
DockSite	Specifies whether the control can be the target of drag-and-dock operations.

Down	Specifies whether the button is selected (down) or unselected (up).
Director	Specifies the Viz Engine director.
DragCursor	Indicates the image used to represent the mouse pointer when the control is being dragged.
DragKind	Specifies whether the control is being dragged normally or for docking.
DragMode	Determines how the control initiates drag-and-drop or drag-and-dock operations. Enable automatic dragging by setting DragMode to dmAutomatic.
DropDownCount	By default, the drop-down list is long enough to contain eight items without requiring the user to scroll to see them all. To make the drop-down list smaller or larger, specify a number larger or smaller than eight as the DropDownCount value. If the DropDownCount value is larger than the number of items in the Items property, the drop-down list will be just large enough to hold all the possible choices and no more. If the DropDownCount value is smaller then the number of item in the Items property, the drop down list will display a scroll bar.
DriverName (Database)	Specifies the name of the BDE driver for the database.
EditMask	Use EditMask to restrict the characters a user can enter into the masked edit control to valid characters and formats. If the user attempts to enter an invalid character, the edit control does not accept the character. Validation is performed on a character-by-character basis by the ValidateEdit method. Setting EditMask to an empty string removes the mask.
EnableBCD	Use the EnableBCD property to specify how fields are mapped to field classes. If EnableBCD is true, (the default value) adDecimal and adNumeric fields are mapped to the TBCDField class when field objects are created. If EnableBCD is false, the fields are mapped to the TFloatField class.
Enabled	Controls whether the control responds to mouse, keyboard, and timer events.
EnableSpellCheck	Enables the user to spell check the text content of a template.
EvObjects	A list of paths to other objects in Viz Engine that should be controlled like the current control.

ExecuteOptions	Set ExecuteOptions to control the characteristics of a command (a data retrieval operation) executed from the ADO data set component.
ExecuteOptions .eoAsyncExecut e	The command or data fetch operation is executed asynchronously.
ExecuteOptions .eoAsyncFetch	The data set first fetches the number of records specified by the CacheSize property synchronously, then fetches any remaining rows asynchronously.
ExecuteOptions .eoAsyncFetchN onBlocking	Asynchronous data fetches or command execution do not block the current thread of execution.
ExecuteOptions .eoExecuteNoRe cords	A command or stored procedure that does not return data. If any rows are retrieved, they are discarded and not returned.
Exclusive (Database)	Enables an application to gain sole access to a database.
Exclusive (Table)	Enables an application to gain sole access to a Paradox or dBASE table.
ExtendedSelect	Determines if the user can select a sequential range of items in the list box.
FieldDefs (DataSet/Table)	Points to the list of field definitions for the data set.
FileDenyBrowse Up	If set when FileStartDir is set, this says that selecting from anywhere else (limiting instead of suggesting) is not permitted.
FileName	The FileName property contains the name and complete directory path of the most recently selected file. The value of FileName is the same as the first item in the Files property.
FileStartDir	If set, this contains the path to the folder (or its subfolders) from which images are selected. For example c:\Users\Public\Pictures\ (final backslash optional).
Filter	The file-selection dialog includes a drop-down list of file types under the edit box. When the user picks a file type from the list, only files of the selected type are displayed in the dialog.

Filter (DataSet/ Query)	Specifies the text of the current filter for a data set.
Filtered (DataSet/Query)	Specifies whether filtering is active for a data set.
FilterOptions (DataSet/Query)	Specifies whether filtering is case insensitive, and whether or not partial comparisons are permitted when filtering records.
FilterIndex	FilterIndex determines which of the file types in Filter is selected by default when the dialog opens. Set FilterIndex to 1 to choose the first file type in the list as the default, or set FilterIndex to 2 to choose the second file type as the default, and so forth.
FindText	FindText contains the text string that the user wants to search for. When the user clicks the Find Next button, the text in the dialog's edit box is automatically assigned to FindText. To make a default text string appear in the edit box when the dialog opens, assign a value to FindText in the Object Inspector or in program code.
Flat	Determines whether the button has a 3D border that provides a raised or lowered look.
FocusControl	Designates a windowed control associated with the label. Set FocusControl to the windowed control that should receive focus when the user presses the accelerator key specified by the label. Specify an accelerator key by preceding a character in the label text with an ampersand (&) and setting the ShowAccelChar property to True.
Font	Controls the attributes of text written on or in the control.
Font.Charset	Specifies the character set of the font.
Font.Color	Specifies the color of the text.
Font.Heigth	Specifies the height of the font.
Font.Name	Identifies the typeface of the font.
Font.Pitch	Specifies whether the characters in the font all have the same width.
Font.Size	Specifies the height of the font in points.
Font.Style	Determines whether the font is normal, italic, underlined, bold, and so on.

ForceDotAsDecimalSeperator	Determines if local specific decimal separator should be ignored and . used as separator.Deprecated - The decimal separator will now always be a dot.
Format	Format specifies custom format string for the date-time display, overriding the control panel strings. A custom format can include both date and time fields, but time fields are not editable unless the Kind property is dtkTime.Examples
Frequency	Use Frequency to specify the spacing of the tick marks, using the logical units used by the Position property. For example, a Frequency of 5 sets a tick mark at every fifth possible increment.
FromPage	The FromPage property corresponds to the From field in the Print dialog.
FullRepaint	Determines how the panel repaints itself when it is resized. When FullRepaint is true, the entire panel, including the beveled border repaints when the size changes. When FullRepaint is false, only the area inside the beveled border repaints.
Global	Determines if the targa clip should be controlled on a global lever or on Viz Engine director level.
Glyph	Specifies the bitmap that appears on the bitmap button.
GroupIndex	Buttons with the same property value (other than 0), work together as a group and can present mutually exclusive choices to the user.
HandleShared (Database)	Specifies whether or not to share a database handle.
Height	Specifies the vertical size of the control in pixels.
HelpContext	HelpContext supports help systems that use numeric topic IDs. For help systems that use topic keywords, see HelpKeyword. To enable context-sensitive help for a control, set HelpType to htContext and set HelpContext to a numeric topic ID. A topic ID of 0 (default) means that no help topic is provided for the control.
HelpKeyword	HelpKeyword supports help systems that use topic keywords. For help systems that use numeric topic IDs, see HelpContext. To enable context-sensitive help for a control, set HelpType to htKeyword and set HelpKeyword to a keyword string. No keyword (zero-length string; default) means that no help topic is provided for the control.

HelpType	Set HelpType to specify whether the control identifies its context-sensitive help topic by ID or keyword. When HelpType is htContext, the HelpContext property identifies the control's help topic. When HelpType is htKeyword, the HelpKeyword property identifies the control's help topic.
HideFileSelector	Determines if the File Selector icon should be hidden.
HideObjects	A list of paths to other objects in Viz Engine that should be hidden if this control is hidden. See HideOnEmpty.
HideOnEmpty	If the current control is empty, meaning there is no text, no image selected etc then the control will hidden in Viz Engine.
HideSelection	Determines whether the visual indication of the selected text remains when focus shifts to another control.
Hint	Contains the text string that can appear when the user moves the mouse over the control.
HorzScrollBar	HorzScrollBar properties.
HorzScrollBar.ButtonSize	Size of the horizontal scroll bar button.
HorzScrollBar.Color	Color of the horizontal scroll bar.
HorzScrollBar.Increment	Scroll increment.
HorzScrollBar.ParentColor	To have a control use the same color as its parent control, set ParentColor to true. If ParentColor is false, the control uses its own Color property.
HorzScrollBar.Position	Indicates the current position of the scroll bar.
HorzScrollBar.Range	Determines how far the scrolling region of the associated control can move.
HorzScrollBar.Visible	Determines if the horizontal scroll bar should be visible or not.
HotTrack	Determines whether labels on the tab under the mouse are automatically highlighted.

ImageInfo	Information about the image selected from Object Store
ImageInfoKey	Deprecated (no longer in use).
ImageKeywords	Keywords for the image selected from Object Store
ImageSources	Use the ImageSources property to set the sources the user should be able to pick images from.
ImageSources.is AdvancedCurio usMap	The user can search and select a map from the Viz World Client.
ImageSources.is Ardome	The user is allowed to search for clips in Ardome.
ImageSources.is Background	Deprecated.
ImageSources.is CuriousMap	The user can search and select a map from the Viz World Client (simplified version).
ImageSources.is ImageMediaSea rch (previously named isDatabase)	The user can select an image from Object Store.
ImageSources.is FileOpen	The user can select a file from the file system.
ImageSources.is Image	The user can drag and drop images from the image pool found under the Viz tab in Director.
ImageSources.is Material	The user can drag and drop materials from the material pool found under the Viz tab in Director.
ImageSources.is Object	The user can drag and drop objects from the "geom" pool found under the Viz tab in Director.
ImageSources.is Paste	The user is allowed to paste images from the clipboard.

ImageSources.is Proximity	The user is allowed to select images or video clips from Proximity's Artbox. Requires the Proximity Artbox asset management system.
ImageSources.is RGB	The user can drag and drop files from the RGB-files tab in Director.
ImageSources.is Scene	The user can drag and drop scenes from the scene pool found under the Viz Engine tab/button in Director.
ImageSources.is VideoMediaSear ch (previously named isVideoHub)	The user can select a video from one of the configured providers in the Media Search.
ImeMode	Determines the behavior of the input method editor (IME). Set ImeMode to configure the way an IME processes user keystrokes. An IME is a front-end input processor for Asian language characters.
ImeName	Specifies the input method editor (IME) to use for converting keyboard input to Asian language characters.
Imported	Boolean telling us if the current was opened and filled externally.
Increment	Determines the increment.
IncrementalDis play	Specifies whether successive approximations of the image should be drawn during slow operations. Set IncrementalDisplay to True on large compressed images that take a long time to load or on images that may be altered by a computationally slow process. When IncrementalDisplay is True, partial renderings of the image are drawn periodically during slow operations such as loading.
IndexDefs (Table)	Contains information about the indexes for a table.
IndexFieldName s (Table)	Lists the columns to use as an index for a table.
IndexFiles (Table)	Specifies one or more dBASE index file(s) to use for a dBASE table that uses non-production indexes.
IndexName (Table)	Identifies a secondary index for the table.

IntegerMode	Determines if the control will only use integer values.
InitialDir	InitialDir determines the default directory displayed in the file-selection dialog when it opens. For example, to point the dialog at the /home/company/docs directory, set the value if InitialDir to /home/company/docs. If no value is assigned to InitialDir, or if the specified directory does not exist, the initial directory is the current working directory.
IntegralHeight	Determines whether the list box displays the partial items.
InternalSpacing	Determines the spacing, in pixels, between each field.
Interval	Interval determines how frequently the OnTimer event occurs. Each time the specified interval passes, the OnTimer event occurs. Use Interval to specify any cardinal value as the interval between OnTimer events. The default value is 1000 (one second).
IsolationLevel	Use IsolationLevel to specify the transaction isolation level for a connection. The transaction isolation level determines how a transaction interacts with other simultaneous transactions when they work with the same tables, and how much a transaction sees of the work performed by other transactions. The default value for IsolationLevel is ilCursorStability.
ItemHeight	Specifies the height in pixels of an item in an owner-draw list box.
ItemsIndex	Indicates which item in the list is selected.
Items	Contains the strings that appear in the list box.
KeepConnection (Database)	Specifies whether an application remains connected to a database even if no data sets are open.
Keyframe	Specifies the Viz Engine keyframe to control.
Kind	Use Kind to specify the appearance of the component's control and its response when the user clicks the button.
Layout	Specifies the vertical placement of the text within the label.
Left	Specifies the horizontal coordinate of the left edge of a component relative to its parent.
Lines	Contains the individual lines of text in the memo control.

LinkName	When selecting an image connected to a person in Object Store, the name of the person can be automatically put into an edit box. Use LinkName to set the edit box that should receive the name of the person
LinkTitle	When selecting an image connected to a person in Object Store, the title of the person can be automatically put into an edit box. Use LinkTitle to set the edit box that should receive the title of the person.
LineSize	Use LineSize to specify the distance along the slider that represents a single unit. The current Position is marked by a slider.
Locked	Determines whether a panel that is used as a toolbar is replaced by a toolbar supplied by an OLE server.
LockType	Use LockType to specify the lock type to use when opening a data set. LockType determines whether the row locking scheme for a record set will be pessimistic, optimistic, or read-only. To be effective, LockType must be set prior to activating the ADO data set component.
LoginPrompt	Specifies whether a login dialog appears immediately before opening a new connection.
Margin	The margin is the space between the image edge and the button edge.
MarshalOptions	Use MarshalOptions when the data set component uses a client-side cursor and data changes are to be sent back to the server. MarshalOptions specifies which records in the local record set are sent to the middle tier or server. A value of moMarshalAll results in all records in the local record set being packaged and sent back. A value of moMarshalModifiedOnly sends back only those records that have actually been changed in the local record set The default value for MarshalOptions is moMarshalAll.
MasterFields (Table)	Specifies one or more fields in a master table to link with corresponding fields in this table in order to establish a master-detail relationship between the tables.
MasterSource (Table)	Specifies the name of the data source for a data set to use as a master table when establishing a detail-master relationship between this table and another one.
Max	Specifies the maximum size of an interval, see Min.
MaxDate	Use MaxDate to get or set the maximum date to which users can scroll the calendar. The values of the Date and EndDate properties cannot exceed MaxDate.

MaxFontSize	Assign a value to MaxFontSize to limit the font sizes available in the dialog. MaxFontSize is inoperative unless the fdLimitSize flag is set in Options.
MaxLength	Specifies the maximum number of characters the user can enter into the edit control.
MaxRecords	Use MaxRecords to control the number of rows the provider for the ADO data set component returns from the data source. Set MaxRecords to indicate the maximum number of rows. The default value for MaxRecords is 0 (zero), which places no limits on the result set.
MaxValue	Specifies the maximum value allowed, see MinValue.
Min	Specifies the minimum size of an interval, see Max.
MinDate	Use MinDate to get or set the minimum date that can be selected.
MinFontSize	Assign a value to MinFontSize to limit the font sizes available in the dialog. MinFontSize is inoperative unless the fdLimitSize flag is set in Options.
MinValue	Specifies the minimum value allowed, see MinValue.
ModalResult	Determines whether and how the button closes its (modal) parent form.
Mode	Read Mode to determine the permissions available to the connection after the connection has been activated. The permissions expressed in Mode govern the types of operations (such as reading and writing) can be performed through the connection. They also control how these operations are affected by and affect other concurrent users.
MultiLine	Determines whether the tabs can appear on more than one row.
MultiSelect	Determines whether the user can select more than one element at a time.
MultiSelection	Indicates if the user is allowed to multi select or not.
Name	Specifies the name of the control as referenced in code.
NumGlyphs	Specifies the bitmap that appears on the bitmap button.
ObjectPath	Reference to the object to control in the Viz Engine scene. Enter the path to the object in the scene. Use full Viz Engine-path or the name of the container with a dollar-sign ($) in front (the container must then have a unique name in the scene tree).

ObjectView (DataSet/Query)	Specifies whether fields are to be stored hierarchically or flattened out in the Fields property.
OEMConvert	Determines whether characters typed in the edit control are converted from ANSI to OEM and then back to ANSI.
Options	Use the Options property to customize the appearance and functionality of the dialog. By default, all options are off.
Options.cdAnyColor	Allow the user to select non-solid colors (which may have to be approximated by dithering).
Options.cdFullOpen	Display custom color options when the dialog opens.
Options.cdPreventFullOpen	Disable the Define Custom Colors button in the dialog, so that the user cannot define new colors.
Options.cdShowHelp	Add a Help button to the color dialog.
Options.cdSolidColor	Direct Windows to use the nearest solid color to the color chosen.
Options.fdAnsiOnly	Displays only fonts that use the Windows character set. Symbol fonts are not listed in the dialog.
Options.fdApplyButton	Displays an Apply button in the dialog, whether or not there is an OnApply event handler.
Options.fdEffects	Displays the Effects check boxes (Strikeout and Underline) and the Color list box in the dialog.
Options.fdFixedPitchOnly	Displays only monospaced fonts in the dialog. Proportionally spaced fonts are not listed, nor are TrueType fonts that do not have the fixed-pitch flag set in the style header.
Options.fdForceFontExist	Allows the user to enter only fonts that are displayed in the dialog (that is, listed in the Font combo box). If the user tries to enter another font name, an error message appears.
Options.fdLimitSize	Enables the MaxFontSize and MinFontSize properties, limiting the range of font sizes that appear in the dialog if these properties have values.

Options.fdNoFaceSel	Causes the dialog to open without a preselected font name in the Font combo box.
Options.fdNoOEMFonts	Removes OEM fonts from the dialog's combo box. Lists only non-OEM fonts.
Options.fdNoSimulations	Displays only fonts and font styles that are directly supported by the font definition file. GDI-synthesized bold and italic styles (for bitmap fonts) are not listed.
Options.fdNoSizeSel	Causes the dialog to open without a preselected size in the Size combo box.
Options.fdNoStyleSel	Causes the dialog to open without a preselected style in the Font Style combo box.
Options.fdNoVectorFonts	Removes vector fonts from the dialog's combo box; lists only non-vector fonts. (Vector fonts are Windows 1.0 fonts, such as Roman or Script, which resemble pen-plotter output.)
Options.fdScalableOnly	Displays only scalable fonts in the dialog. Non-scalable (bitmap) fonts are removed from the list.
Options.fdShowHelp	Displays a Help button in the dialog.
Options.fdTrueTypeOnly	Displays only TrueType fonts in the dialog. Other fonts are not listed.
Options.fdWysiwyg	Displays only fonts that are available to both the printer and the screen. Device-specific fonts are not listed in the dialog.
Options.frDisableMatchCase	Disables (grays) the Match Case check box in a find dialog.
Options.frDisableUpDown	Disables (grays) the Up and Down buttons, which determine the direction of the search.
Options.frDisableWholeWord	Disables (grays) the Match Whole Word check box of find dialog.
Options.frDown	Selects the Down button by default when the dialog opens. If the frDown flags is off, Up is selected when the dialog opens. (By default, frDown is on.)

Options.frFindNext	This flag is turned on when the user clicks the Find Next button and turned off when the dialog closes.
Options.frHideMatchCase	Removes the Match Case check box from the dialog.
Options.frHideUpDown	Removes the Up and Down buttons from the dialog.
Options.frHideWholeWord	Removes the Match Whole Word check box from the dialog.
Options.frMatchCase	This flag is turned on (off) when the user selects (deselects) the Match Case check box. To select the check box by default when the dialog opens, set frMatchCase at design time.
Options.frReplace	Applies to TReplaceDialog only. This flag is set by the system to indicate that the application should replace the current occurrence (and only the current occurrence) of the FindText string with the ReplaceText string. Not used in search routines.
Options.frReplaceAll	Applies to TReplaceDialog only. This flag is set by the system to indicate that the application should replace all occurrences of the FindText string with the ReplaceText string.
Options.frShowHelp	Displays a Help button in the dialog.
Options.frWholeWord	This flag is turned on (off) when the user selects (deselects) the Match Whole Word check box. To select the check box by default when the dialog opens, set frWholeWord at design time.
Options.ofAllowMultiSelect	Allows users to select more than one file in the dialog.
Options.ofCreatePrompt	Generates a warning message if the user tries to select a nonexistent file, asking whether to create a new file with the specified name.
Options.ofDontAddToRecent	Prevents the file from being added to the list of recently opened files.
Options.ofEnableIncludeNotify	(Windows 2000 and later.) Sends CDN_INCLUDEITEM notification messages to the dialog when the user opens a folder. A notification is sent for each item in the newly opened folder. It is also possible to use these messages to control which items appear in the folder's item list.

Options.ofEnabl eSizing	(Windows 98 and later) Lets the Explorer-style dialog be resized with the mouse or keyboard. By default, the dialog allows this resizing regardless of the value of this option. It is only required when a hook procedure or custom template is provided (old style dialogs never permit resizing).
Options.ofExten sionDifferent	This flag is turned on at runtime whenever the selected filename has an extension that differs from DefaultExt. Remember to reset it if this flag is used in an application.
Options.ofFileM ustExist	Generates an error message if the user tries to select a nonexistent file. (only applies to Open dialogs).
Options.ofForce ShowHidden	Ensures that hidden files are visible in the dialog.
Options.ofHide ReadOnly	Removes the Open As Read Only check box from the dialog.
Options.ofNoCh angeDir	After the user clicks OK, resets the current directory to whatever it was before the file-selection dialog opened.
Options.ofNoDe referenceLinks	Disables dereferencing of Windows shortcuts. If the user selects a shortcut, assigns to FileName the path and file name of the shortcut itself (the .LNK file), rather than the file linked to the shortcut.
Options.ofNoLo ngNames	Displays 8.3-character file names only. This option is only valid if Options also includes ofOldStyleDialog.
Options.ofNoNe tworkButton	Removes the Network button (which opens a Map Network Drive dialog) from the file-selection dialog. Applies only if the ofOldStyleDialog flag is on.
Options.ofNoRe adOnlyReturn	Generates an error message if the user tries to select a read-only file.
Options.ofNoTe stFileCreate	Disables checking for network file protection and inaccessibility of disk drives. Applies only when the user tries to save a file in a create-no-modify shared network directory.
Options.ofNoVa lidate	Disables checking for invalid characters in file names. Allows selection of file names with invalid characters.
Options.ofOldSt yleDialog	Creates the older style of file-selection dialog.

Options.ofOver writePrompt	Generates a warning message if the user tries to select a file name that is already in use, asking whether to overwrite the existing file. (Use with save dialogs.)
Options.ofPath MustExist	Generates an error message if the user tries to select a file name with a nonexistent directory path.
Options.ofRead Only	Selects the Open As Read Only check box by default when the dialog opens.
Options.ofShare Ware	Ignores sharing errors and allows files to be selected even when sharing violations occur.
Options.ofShow Help	Displays a Help button in the dialog.
Options.poDisa blePrintToFile	Disables (grays) the Print To File Check box. (Applicable only if the poPrintToFile flag is set.)
Options.poHelp	Displays a Help button in the dialog. May not work in some versions of Windows 95.
Options.poPage Nums	Enables the Pages radio button, allowing the user to specify a page range.
Options.poPrint ToFile	Displays a Print To File check box in the dialog.
Options.poSelec tion	Enables the Selection radio button, allowing the user to print selected (highlighted) text.
Options.poWarn ing	Generates a warning message if the user tries to send a job to an uninstalled printer.
OptionsEx	Use the OptionsEx property to further customize the file open dialog beyond the options covered by the Options property.
OptionsEx.ofEx NoPlacesBar	Setting ofExNoPlacesBar to True hides the "Places Bar". The "Places Bar" provides shortcuts to places on the computer or the network from which a file can be opened, such as the History folder, the desktop, or My Network Places.
Orientation	Specifies whether the track bar is horizontal or vertical.

OwnerDraw	Set OwnerDraw to true to enable events that allow the control to draw itself by overriding any default painting behavior.
PageSize	Specifies the amount Position is changed when the PageUp or PageDown key is pressed, or when the bar is clicked.
ParamCheck (Query)	Specifies whether the parameter list for a query is regenerated if the SQL property changes at runtime.
Parameters	At design-time, use Parameters to access the parameters for the SQL statement used by the ADO component. Use the properties and methods of Parameters to set and get individual parameter values or to set or inspect parameter attributes. At runtime, use the Parameters property editor to access the collection of parameters.
Params (Database)	Contains database connection parameters for the Borland Database Engine (BDE) alias associated with the database component.
Params (Query)	Contains the parameters for a query's SQL statement.
ParentBackground	Determines whether control uses parent's theme background. If ParentBackground is true, the control uses the parent's theme background to draw its own background. ParentBackground has no effect unless XP themes are enabled.
ParentBiDiMode	Specifies whether the control uses its parent's BiDiMode.
ParentColor	Determines where a control looks for its color information. To have a control use the same color as its parent control, set ParentColor to True. If ParentColor is False, the control uses its own Color property.
ParentCtl3D	Determines where a component looks to determine if it should appear three dimensional.
ParentFont	Determines where a control looks for its font information. To have a control use the same font as its parent control, set ParentFont to True. If ParentFont is False, the control uses its own Font property.
ParentShowHint	Determines where a control looks to find out if its Help Hint should be shown.
ParseInput	Enables the OnUserInput event. If ParseInput is true, the OnUserInput event occurs whenever the user types into the component's edit box.
PasswordChar	Indicates the character, if any, to display in place of the actual characters typed in the control.

PersonInfo	List with information about the person selected from Object Store.
picFilename	Contains the path to the image/map/video that the control contains.
Picture	Specifies the image that appears on the image control.
Pool	The Viz Engine pool that the item selector should display (like SCENE, GEOM, FONT etc).
PopupMenu	Identifies the pop-up menu associated with the control.
Position	Contains the current position of the slider of a TTrackBar.
Prefix	Specifies the prefix for a save name.
Prepared	Set Prepared before calling the Open method to specify whether ADO prepares the command used to create the data set's record set. If Prepared is set to true and the data set component is connected to a data store, ADO prepares the command before executing it. If Prepared is set to false, ADO does not prepare the command. The default value of Prepared is false.
PrintRange	The value of PrintRange corresponds the All, Selection, and Pages (From/To) radio buttons in the Print dialog.
PrintToFile	PrintToFile is true whenever the Print To File check box is selected in the dialog.
Proportional	Indicates whether the image should be changed, without distortion, so that it fits the bounds of the image control.
Provider	Set Provider to indicate the provider the ADO connection object uses. The provider a connection is to use can also be specified in the string entered in ConnectionString. If Provider is left blank and the provider specified through ConnectionString, when the connection is activated the Provider property is automatically set indicating the provider used.
RaggedRight	Specifies whether rows of tabs stretch to fill the width of the control.
RDSConnection	Use RDSConnection to specify the RDS connection component to use to obtain the record set when the TADODataSet is opened. This property is mutually exclusive of the Connection property.
ReadOnly	Determines whether the user can change the text of the edit control.

ReadOnly (Database)	Specifies that the database connection provides read-only access.
ReplaceText	ReplaceText, taken from the Replace With edit box, contains the text string that will replace FindText.
RequestLive (Query)	Specifies whether an application expects to receive a live result set from the Borland Database Engine (BDE) when the query executes.
RequiredKeyWords	Defines the required keywords used when searching for media content. This is currently relevant for Object Store searches only.
RequiredKeyWordsMode	Sets the options RequireOne or RequireAll. Searches can be performed with information from the RequiredKeyWords field.
RTL	Right to left reading.
RTL.rtlReading	Specifies that the text is displayed with right to left reading order.
RTL.rtlReading	The text is right-aligned.
ReverseSorting	Indicates if reverse sorting is active.
SaveNameTextFields	Click the ellipsis button to bring up and editor where you can select the edit boxes that should be used for generating a save name.
ScrollBars	Determines whether the memo control has any scroll bars.
ScrollOpposite	Determines how the rows of tabs are scrolled in a multiline tab control.
ScrollWidth	Specifies the width, in pixels, by which the list box can scroll horizontally.
SelEnd	Specifies the position of the end point of the selection range.
SelectedPath	The Viz Engine path of the selected item.
SelStart	Specifies the position of the starting point of the selection range.
SendSJIS	Sends the text in the text box as SJIS (character encoding for the Japanese language).
SendUTF8	Sends the text in the text box as UTF-8 (universal character encoding).

Seperator	Specifies the size of the tabs in e.g. a list box or page control.
SessionName (Query/ Database)	Identifies the name of the session with which this data set is associated.
ShowAccelChar	Determines how an ampersand (&) in the label text is displayed. Set ShowAccelChar to True to allow the label to display an underlined accelerator key value. When ShowAccelChar is True, any character preceded by an ampersand appears underlined. If the FocusControl property is set, the windowed control specified by the FocusControl property receives input focus when the user types that underlined character. To display an ampersand when ShowAccelChar is True, use two ampersands (&&) to stand for the single ampersand that is displayed. Set ShowAccelChar to False to display the label text with all ampersands appearing as ampersands. When ShowAccelChar is False, the value of the FocusControl property is not used.
ShowHint	Determines whether the control displays a Help Hint when the mouse pointer rests momentarily on the control.
ShowProximityS elector	Specifies if the control should show a Proximity selection button.
ShowSaveDialog	Determines whether Director should show the save dialog when saving the template.
ShowTimeStam p	Indicates if the control will show timestamp.
SliderVisible	Specifies whether the slider should be visible or not.
SortByDate	Sort entries by date.
SortByName	Sort entries by name.
Sorted	Specifies whether the items in a list box or combo box are arranged alphabetically.
Source	Deprecated (no longer in use).
Spacing	Spacing determines the number of pixels between the image (specified in the Glyph property) and the text (specified in the Caption property). The default value is 4 pixels.
SQL (query)	Contains the text of the SQL statement to execute for the query.

State	Indicates whether the check box is selected, deselected, or grayed.
StoreDefs (Table)	Indicates whether the table's field and index definitions persist with the data module or form.
StoryActions	Click the ellipsis button to bring up and editor where you can select and order the edit boxes that should get the data we receive externally.
Stretch	Indicates whether the image should be changed so that it exactly fits the bounds of the image control.
Strings	List of strings.
Style	Determines the style of a list box or combo box.
TableDirect	Use TADOTable to access data in a single database table using ADO. TADOTable provides direct access to every record and field in an underlying database table. An ADO table component can also work with a subset of records within a database table using ranges and filters.
TableName	Use TableName to specify the base table in a database on which the ADO table component operates. Data is retrieved from and changes written to this base table. Before setting or changing the value of TableName, deactivate the ADO table component by calling its Close method or setting its Active property to false. At design-time, if a valid connection to a database can be established, the Object Inspector provides a drop-down list of names of available tables.
TableType (Table)	Indicates the database table structure for the table that this component represents.
TabHeight	Specifies the size of the tabs in e.g. a list box or page control.
TabIndex	Specifies the order of the tabs in e.g. a list box or page control.
TabOrder	Indicates the position of the control in its parent's tab order. TabOrder is the order in which child windows are visited when the user presses the Tab key. In most applications, the *control* with the TabOrder value of 0 is the control that has the focus when the form first appears. In Pilot News however, it is the *text box* with the smallest TabOrder that has the initial focus.
TabPosition	Specifies the position of the tabs in e.g. a list box or page control.
TabStop	Determines if the user can tab to a control.

TabWidth	Specifies the size of the tabs in e.g. a list box or page control.
Tag	Stores an integer value as part of a component. Tag has no predefined meaning. The Tag property is provided for the convenience of developers. It can be used for storing an additional integer value or it can be typecast to any 32-bit value such as a component reference or a pointer.
Text	Contains a text string associated with the control.
ThumbLength	Specifies the size of the slider.
TickMarks	Specifies where the slider ticks are shown.
TickStyle	Specifies how tick marks are placed on the track bar.
Title	Use Title to return or set the title caption for the dialog box.
Top	Specifies the Y coordinate of the top left corner of a control, relative to its parent or containing control in pixels.
ToPage	The ToPage property corresponds to the To field in the Print dialog.
TransIsolation (Database)	Specifies the transaction isolation level for transactions managed by the Borland Database Engine (BDE).
Transparent (Label)	Specifies whether controls that sit below the label on a form can be seen through the label.
Transparent	Specifies whether the background of the image obscures objects below the image object.
TWImageInfCo mponent	Deprecated (no longer in use).
UHPathIsRelativ e	If true, then the path in UpdateHubPath is considered relative to the current position in the UpdateHub tree. Otherwise, it is treated as an absolute path.
UpdateHubPath	The path in the UpdateHub that control edits or displays.
UTF8Items	The items in the control represented as a list of UTF-8 strings.
UTF8Text	Defines the character encoding. UTF-8 is a variable-length character encoding for Unicode, representing any universal character in the Unicode standard, and contains virtually all characters in common use.

UniDirectional (Query)	Determines whether or not Borland Database Engine (BDE) bidirectional cursors are enabled for a query's result set.
UnixShare	Deprecated (no longer in use).
UpdateMode (DataSet/ Query/Table)	Determines how the Borland Database Engine (BDE) finds records when updating to an SQL database.
UpdateObject (DataSet/Query)	Specifies the update object component used to update a read-only result set when cached updates are enabled.
UseDockManager	Use UseDockManager to get or set whether a docking manager is used in drag-and-dock operations.
UseUTF8UnicodeProperty	When the DirectStoryImporter receives data and sets text into the various text boxes, this property determines if it will use the .text or the .utf8text property of the edit fields.
Value	Specifies the value.
VertScrollBar	Vertical scrollbar properties.
VertScrollBar. ParentColor	To have a control use the same color as its parent control, set ParentColor to true. If ParentColor is false, the control uses its own Color property.
VertScrollBar.Position	Indicates the current position of the scroll bar.
VertScrollBar.Range	Determines how far the scrolling region of the associated control can move.
Visible	Determines whether the component appears on screen.
VizDenyBrowseUp	Indicates if the user should be allowed to navigate up in the Viz Engine tree when selecting a Viz resource in an image control. Note that VizDenyBrowseUp is ignored in Director. This means that while users of Viz Pilot News inside a Newsroom System are limited by the VizDenyBrowseUp property, users of Director are considered graphics/video/maps super users and can use any image in the Director template.
VizStartDir	Starting location in the Viz Engine tree when selecting a Viz resource in an image control. Only considered if there is no pre-selected value in the control and, in Pilot News, also when VizDenyBrowseUp is set and the pre-selected value is not within the folder specified here.

VizText	Deprecated (no longer in use).
WantReturns	Determines whether the user can insert return characters into the text.
WantTabs	Determines whether the user can insert tab characters into the text.
Width	Specifies the horizontal size of the control or form in pixels.
WordWrap	Determines whether the edit control inserts soft carriage returns so text wraps at the right margin.
XValue	Value for X.
YValue	Value for Y.
ZValue	Value for Z.

10.8   Import Components

Import components are used when importing playlists from newsroom systems and when creating macros to automatically fill templates. Imports can be executed manually or automatically with the use of scripts and macros.

It is good practice for scene designers to name the scene's Field Identifiers using 01, 02, and 03 rather than 1,2,3,4 … 11,12 and so on. Using 1,2,3,4 will cause data to be assigned to the wrong fields when using Import components since the Media Sequencer will place 11 and 12 before 2 and so on.

This section contains the following topics:

- Story Importer Component
- Direct Story Importer Component

10.8.1   Story Importer Component

Using TStoryImporter, a template can be either manually or automatically filled using a macro or by importing an external data source (XML).

When filling a template, a template is specified with an External ID and data is added to the template. The order in which the data is filled is defined by the order of the ControlObjectName property which refers to the Field Identifier set by the scene designer.

This section contains the following topics:

- External ID
- XML Example
- Script Example
- Creating a Macro Using External IDs in Director
- Creating Data Elements Based On External IDs
- Using a Script to Control Data Filling in the Template

External ID
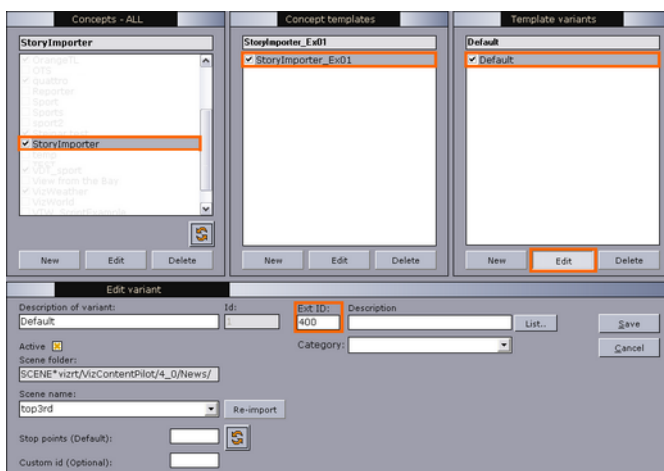
When created, a template is saved and typically added to a concept. In Template Manager, the template can be given an external ID (see Variants) for use with Import components.

This ID is then used when creating data elements by triggering macros in Director, adding playlists to Director manually from newsroom systems (see Create a macro using External IDs in Director). External IDs are also used with newsroom systems that create data elements based on *cg* commands; however, Import component does not support this workflow.

XML Example

The XML example below contains the name of the playlist, a group and three elements with an External ID (`templateID`). The XML playlists are generated by newsroom systems, and manually imported by Director's XML Filler.

```xml
<?xml version="1.0"?>
<playlist>
    <name>Name of playlist</name>
    <group visible="true" name="Group name">
        <event createType="CreateDataElement">
            <templateID>110</templateID>
            <dataName>Data Element 001</dataName>
            <comment/>
            <data>
                <field>※</field>
                <field>آخر النهر</field>
                <field>Star</field>
            </data>
        </event>
        <event>
            <templateID>110</templateID>
            <dataName>Data Element 002</dataName>
            <comment/>
            <data>
                <field>ؼ</field>
                <field>Pleione</field>
                <field>Binary star</field>
            </data>
        </event>
        <event createType="CreateDataElement">
            <templateID>110</templateID>
            <dataName>Data Element 003</dataName>
            <comment/>
            <data>
                <field>❖</field>
                <field>Ταὐρος</field>
                <field>Constellation</field>
            </data>
        </event>
    </group>
</playlist>
```

Script Example
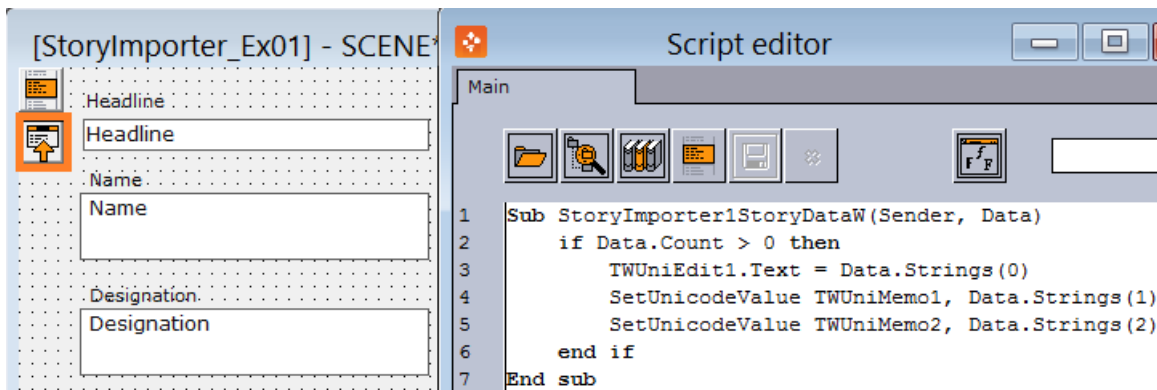
The component triggers its OnStoryDataW event to start filling the template with Data.

```
Sub StoryImporter1StoryDataW(Sender, Data)
    if Data.Count > 0 then
        TWUniEdit1.Text = Data.Strings(0)
        SetUnicodeValue TWUniMemo1, Data.Strings(1)
        SetUnicodeValue TWUniMemo2, Data.Strings(2)
    end if
End sub
```

⚠ **Note:** From Viz Pilot 8.2, **OnStoryDataW** should be used instead of the old OnStoryData event, as **OnStoryDataW** handles Unicode better.

Creating a Macro Using External IDs in Director

1. Start **Director**.
2. On the **Options** menu select **Keyboard Configuration…** .
3. Select and right-click the **macros** category, and from the context menu select **Add Macro…** .
4. Click **Show Macros…** to open the **Predefined Functions** window.
5. Select the **create_data_element** function under **templates**.
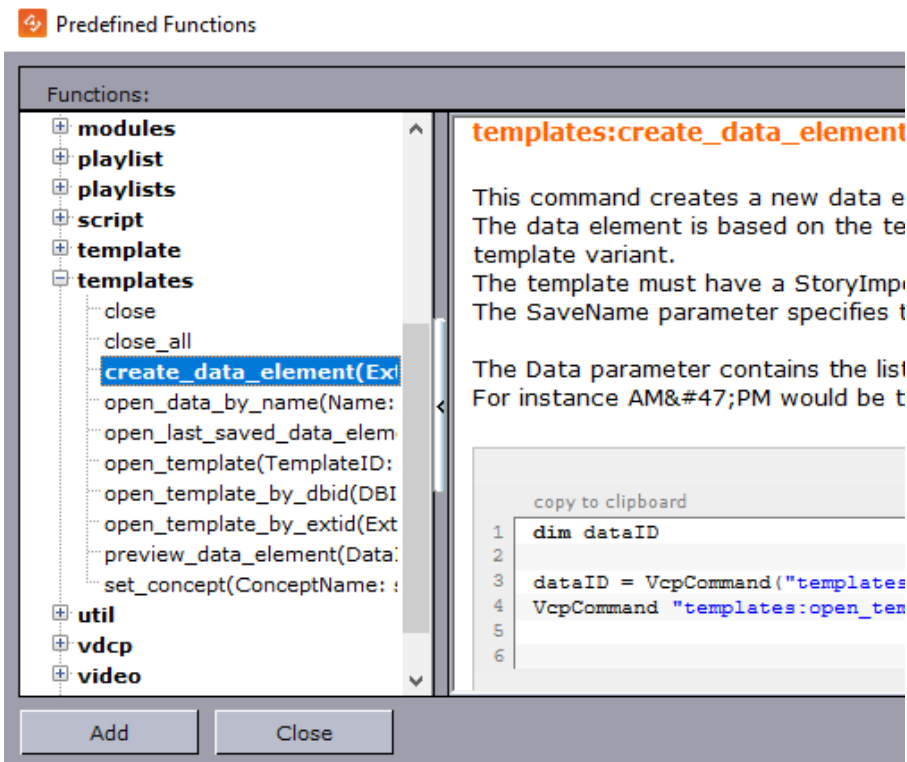
6. Click the hyperlink named **copy to clipboard**, and click **Close**.
7. In the **New Macro Shortcut** window select the **VBScript** tab, and paste the newly copied script into the script editor.
8. Edit the script.

```
dim dataID
dataID = VcpCommand("templates:create_data_element 400
MyNewElement Headline/Title/Designation")
VcpCommand "templates:open_template -1 " & dataID
```

> ⚠️ **Note:** The number 400 represents the external ID added in Template Manager. MyNewElement is the name of the template, and Headline, Title and Designation are data that will be filled into the template by the template script.

9. Select the **Enter Key** field, and press the key combination for the macro.
10. Enter a **Macro Name**, and click **OK**.

Creating Data Elements Based On External IDs

1. In **Director** press the newly assigned key combination to trigger the macro (see Script Example).
2. Enter the information required.
3. Click the **Save** button to save it as a data element.
4. Add the data elements to the playlist.

Using a Script to Control Data Filling in the Template
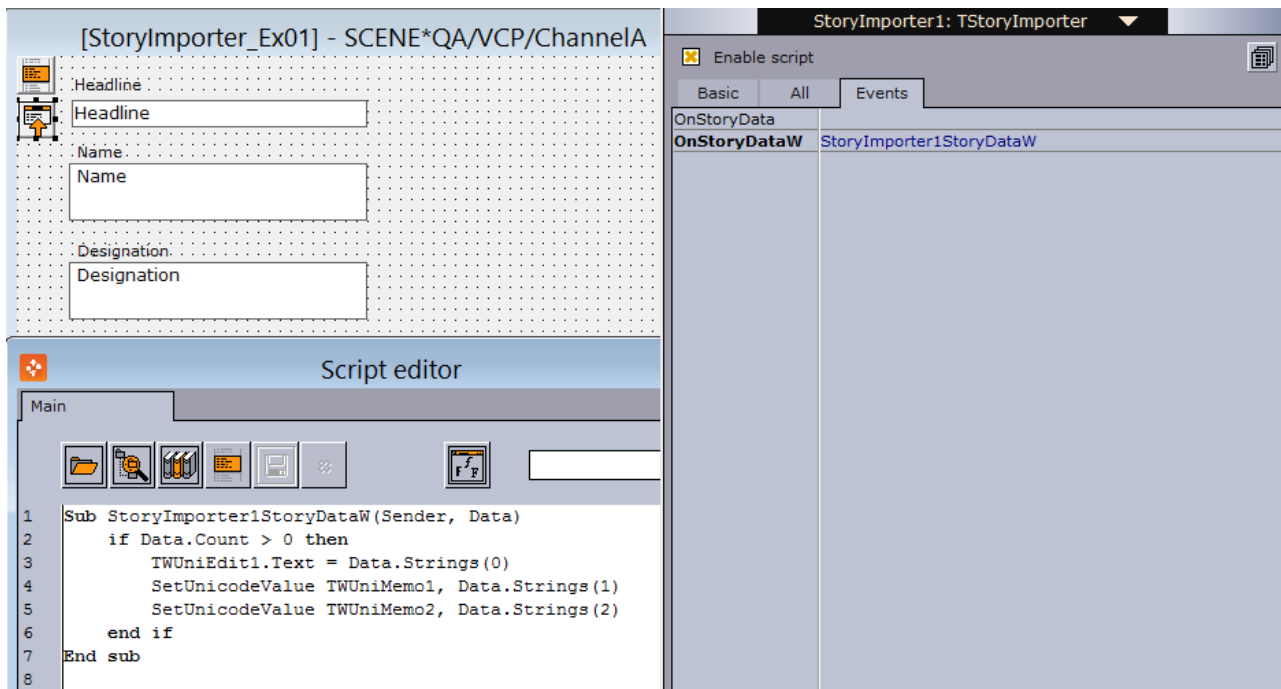
1. In **Template Wizard**, open the Object Inspector and check the **Enable script** option to enable scripting (**CTRL+F12**).
2. Add the TStoryImporter component to the template (select the Import category from the dropdown).
3. Select the TStoryImporter component in the template. Click Events in the object inspector and double click the **OnStoryDataW** event.
4. Set the body of the newly created event handler like this:

```
Sub StoryImporter1StoryDataW(Sender, Data)
    if Data.Count > 0 then
        TWUniEdit1.Text = Data.Strings(0)
        SetUnicodeValue TWUniMemo1, Data.Strings(1)
        SetUnicodeValue TWUniMemo2, Data.Strings(2)
    end if
End sub
```

⚠️ **Note:** From Viz Pilot 8.2, **OnStoryDataW** should be used instead of the old OnStoryData event, as **OnStoryDataW** handles Unicode better.

10.8.2 Direct Story Importer Component

The TDirectStoryImporter component is used for fetching template data from external sources such as XML files. This is similar to the Story Importer Component; however, the Direct Story Importer component is different in that it has an editor for mapping template text fields to index IDs, and is therefore easier to use without scripting. If advanced scripting is needed, the Story Importer Component component is recommended.

To understand the logic and use of the component in Director, see Story Importer Component.

Notable Properties

- **StoryActions**: Opens the Story Actions Editor for linking text fields to index values.

Using the TDirectStoryImporter Component

1. Add the **TDirectStoryImporter** component from the `Import` component palette to the template.
2. Click the ellipse (**...**) button to open the **Story Actions Editor**.
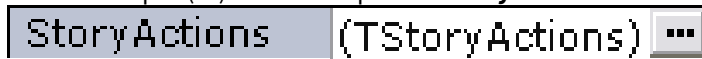
3. Link edit fields to index values so that they correspond with the sequence of data coming from the external system.
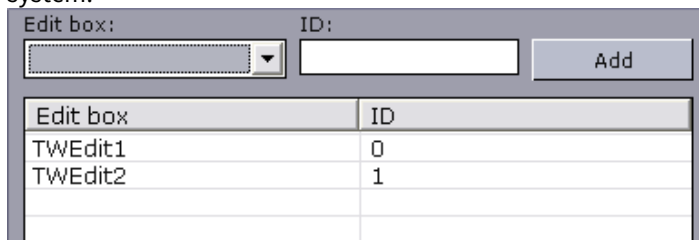
4. Click **OK**.
5. Save the template.

10.9   Standard Components

Standard components are the most commonly used components. This section contains the following topics:

- Unicode Label Component
- Unicode Edit Component
- Unicode Memo Component
- Image Component
- World Maps Editor Component
- Panel Component
- Unicode Button Component
- Unicode CheckBox Component
- Unicode Radio Button Component
- Unicode ComboBox Component
- Unicode ListBox Component
- Unicode GroupBox Component
- Static Image Component
- Page Control Component
- Template Information Component
- Person Information Component
- Feed Linker Component

10.9.1   Unicode Label Component

Label adds a basic text label to the template.

Notable Properties

- **Caption:** Changes the label text and sets the caption.
- **Font:** Sets the font type and size.

Script Example

TWUniLabel1

```
TWUniLabel1.Font.Color = clRed
TWUniLabel1.Caption = "Special chars çgiösü æøå " + ChrW(&H25B3)
```

10.9.2   Unicode Edit Component

Edit adds a basic text field editor to the template.

Notable Properties

- **EnableSpellCheck:** Enables spell checking for the text field.
- **Name:** Sets the name of the text field.
- **Text:** Sets the text in the text field (recommended from Viz Pilot 8.3 and later).
- **Utf8Text:** The recommended way of setting text prior to Viz Pilot 8.2.

Script Example

Please see below for the results of setting unicode text in the different properties. Below, *ChrW(&H25B3)* should add a triangle.

```
Sub testingclick(Sender)
  TWUniEdit1.Text = "çgiösü æøå " + ChrW(&H25B3)
  TWUniEdit2.Utf8Text = "çgiösü æøå " + ChrW(&H25B3)
  SetUnicodeValue TWUniEdit3, "çgiösü æøå " + ChrW(&H25B3)
End sub
```

Viz Pilot before version 8.2:

Viz Pilot 8.3 and later, with the **legacy_unicode_handling** setting enabled (for templates written before Viz Pilot 8.2 using the Utf8Text property):

Viz Pilot 8.3 and later:
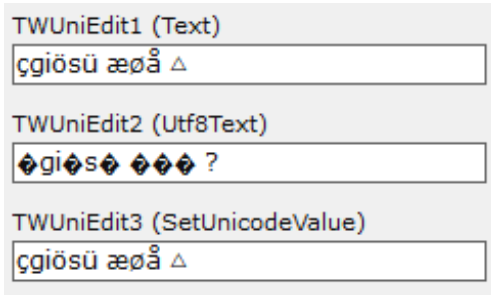
Please note, the goal has been to support UTF8 and emojis and special characters (like the triangle above) properly, in the **Text** property. Unfortunately this broke the **Utf8Text** property.

The text field should be **set** as follows:

- `TWUniEdit1.Text = Date` or,
- `SetUnicodeValue TTWUniEdit1, Date`

Another example using the Unicode Edit field is described in this Script Example (Image with Name, Unicode Edit and Template Info).

Enabling Spell Checking in Unicode Edit and Unicode Memo Fields

1. Use the Wizard to create a new template.
2. Add a Unicode Edit Component or Unicode Memo Component field to the template.
3. Check the **EnableSpellCheck** property for each field that requires spell checking.
4. Save the template.

> ⚠️ **Note:** Spelling is only checked when the template is saved as a data element in the Newsroom System or Director.

10.9.3   Unicode Memo Component

Memo is a wrapper for a multiline edit control. Multiline edit boxes let the user enter more than one line of text. They are appropriate for representing lengthy information.

> ❌ **Caution:** When a text memo component is created manually and the user creates a data element from the template, a new line is added automatically to the content. This does not happen when the text memo is created automatically from a scene by Template Wizard.

Notable Properties

- **EnableSpellCheck:** Enables spell checking for the text field.
- **Name:** Sets the name of the multiline text field.
- **Text:** Sets the text in the text field.

- **WordWrap:** Wraps a section of words, adding line breaks (this property is listed in the **All** tab). Refer to the example below for more information.

Script Example
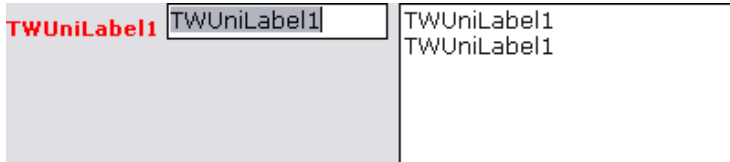

```
TWUniLabel1.Font.Color = clRed
TWUniEdit1.Text = TWUniLabel1.Caption
TWUniMemo1.Text = TWUniLabel1.Caption & vbCrLf & TWUniLabel1.Caption
```

⚠️ **Note:** The code snippet `vbCrLf` __adds a line break after the first text element.

When enabling the *WordWrap* property, the width of the template's text box determines where the line breaks in the graphics will be. The output on the renderer might therefore not be as intended. If the template's text box is narrower than what the scene allows, the output will have line breaks according to the text box, and not according to the scene design. To solve this, either use the *Wrap Fixed Unicode Memo* component without *WordWrap*, or add a script that can remove all line breaks. The example below is most useful for scenes that only contain one text paragraph.

```
temp1 = TWUniMemo1.Text
temp1 = replace(temp1, chr(10), "")
temp1 = replace(temp1, chr(13), "")
TWUniMemo1.UTF8Text = temp1
```

✅ **Tip:** Use the Wrap Fixed Unicode Memo Component to avoid scripting.

10.9.4   Image Component

The **TTWImage** component (also known as **Image with Name and Title Linking Component**) has several uses. In addition to being used for setting images in Viz Engine scenes, it can also be configured to control backgrounds, objects and materials. It also has a special feature that lets users add a video instead of an image.

Notable Properties

- **AllowCropAspectOverride:** Lets users select a different aspect ratio than the one given by the DestHeight and DestWidth properties, when using Crop Service. Default value = False. See Preset Image Size.

- **AlwaysUseCropTool:** Lets users always use the crop tool by default when an image is added to the template when using the isImageMediaSearch, isFileOpen and isPaste image source options. See Enable the crop tool.
- **ControlObjectName:** Links the component to an exposed scene object.
- **DestWidth/DestHeight:** Sets the destination width/height of the image, forcing the size of the image used in the scene to be of a specific width/height. For details, see Preset Image Size.
  When set to values greater than 0 (zero), it will enable the crop tool. The parameters set a fixed height and width for the crop tool. Using this setting will force the user to use the configured height and width, and the aspect. Meaning, if the selection is greater than then actual height and width parameters, the image will be scaled to fit.
- **DisallowedKeywords:** Defines the disallowed keywords used when searching for media content. When users searches and finds an image, the image cannot be selected if it contains a keyword that is present in the DisallowedKeywords property.

> ⚠️ **Note:** The DisallowedKeywords property is available for RestVOS and Viz One searches, but not for Object Store searches.

- **EvObjects:** Add the path to more object containers to receive the same value as the object defined in ObjectPath. Add the name of containers with "$containername" or the full Viz Engine path. Other objects must have the same keyframe name as the main object.
- **HideObjects:** Define other objects that should be hidden when the main image object is hidden.
- **HideOnEmpty:** If selected, the object is set to hidden in the scene if no image is loaded.
- **ImageSources:** Sets one or multiple image source locations. The default settings are *isImage* and *isImageMediaSearch*; however, in most cases the actual control object properties of the scene will decide what components and settings are used.
    - **isAdvancedCuriousMap:** Lets the user search and select a map from the Viz World Client using the Classic Edition Editor. (*isAdvancedCuriousMap* property is deprecated from Viz Pilot 5.7, please use World Maps Editor Component instead.)
    - **isArdome:** Requires an Ardome connection and setup to let the user search and select clips from Ardome.
    - **isCuriousMap:** Lets the user search and select a map from the Viz World Client using the Lite Edition Editor. (*isCuriousMap* property is deprecated from Viz Pilot 5.7 and the Lite Edition Editor is not the recommended editor, please use World Maps Editor Component instead.)
    - **isFileOpen:** Lets the user search and select an image from disk. Note that the image must be available on the Viz Engine using the same shared path.
    - **isImage:** Lets the user search the Viz *image pool* for images through Director's *Viz* resource tab. Viz 3.x images are stored on the Viz Graphic Hub while Viz images are stored on its data root.
    - **isImageMediaSearch** (previously named **isDatabase**): Lets the user search and select a still image from the configured service providers (for example Object Store, Viz One).
    - **isMaterial:** Lets the user select a material from the Viz *material pool* through Director's *Viz* resource tab.
    - **isObject:** Lets the user select an object from the Viz *object pool* through Director's *Viz* resource tab (requires a local Viz Engine connection).
    - **isPaste:** Lets the user paste images from the Windows clipboard by right-clicking the Image icon and selecting Paste... on the context menu. Pasting will open the Crop Image tool. Pasted images should be added to a shared drive accessible to Viz Engine. The crop tool save path is set in Director's

Preferences window under General settings. isPaste does not add images to Object Store, and does not preserve alpha values.
- **isProximity:** Lets the user search and select images or clips from Proximity's Artbox. Requires the Proximity Artbox asset management system.
- **isRGB:** Lets the user select an image from the Viz *RGB pool* through Director's *Viz* files tab (Viz Pilot News cannot use *isRGB*, as this relates to a Director specific function).
- **isScene:** Lets the user select a scene from the Viz *scene pool* through Director's *Viz* resource tab.
- **isVideoMediaSearch** (previously named **isVideoHub**): Lets the user search and select a video from the configured service providers (for example Object Store, Viz One). When selected, it lets the user of the template preview a video item using the Timeline Editor within Director or Viz Pilot News.

- **RequiredKeyWords:** Defines the required keywords used when searching for media content. See Required Keywords.
- **RequiredKeywordsMode:** Sets the options `RequireOne` or `RequireAll` . Searches can be performed with information from the RequiredKeyWords field.

Events

- **OnImageInfoChanged:** Enables Object Store to Add person information to the template. Person information is often registered in Object Store with images of persons in order to easily retrieve the correct name of the person in question. See Person Information Component.
- **CreatingMediaOrder:** Used with Order Management. This event is raised just before the Order Management window is shown. Its signature looks like:

```
Sub TWImageInf1CreatingMediaOrder(Sender, CustomParameters,
ReservedParameterKeys)
```

- `Sender` is set to the image control itself.
- `CustomParameters` is a key-value collection where you can add items. For example:
  `CustomParameters.Add("this=1")`
- `ReservedParameterKeys` is a list containing all the keys that are set by the client for various purposes. It has the following values: keywords, asset_type, destination_width, destination_height, template_name, concept_name, variant_name. If any of these keys is set in `CustomParameters` then it will be ignored and the value provided by the client will be used instead.
  For **example:**

```
Sub TWImageInf1CreatingMediaOrder(Sender, CustomParameters,
ReservedParameterKeys)
    ' Add key value parameter pair
    CustomParameters.Add "Param1=Value1"
    ' Get index of key
    CustomParameters.IndexOfName("Param1") ' returns 0 (-1 if name is not
found)
    ' Get the key for a given index
    CustomParameters.Names(o) ' returns "Param1"
```

```
        ' Get the value for a key
        CustomParameters.Values( "Param1" ) ' returns "Value1"
    End sub
```

Script Example (Image with Name, Unicode Edit and Template Info)

The following example describes how to include the Object Store image name in the data element name.
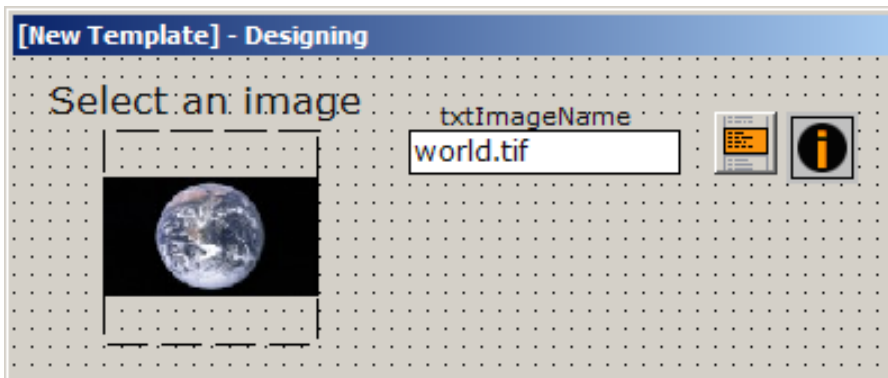
1. Add a Unicode Edit Component field (name `txtImageName`) and set it to be invisible.

2. Add a Template Information Component component and add in `txtImageName` as one of the save name fields.

3. On any of the image components (here named " `image` ") add the following script on the `OnFilenameChanged` event:

```
Sub imageFilenameChanged(Sender)
    txtImageName.UTF8Text = "" 'Clear the save name before starting
\qFirst test if the returned image is from a Viz One server
    if Left(image.picFilename, 8) = "https://" then
        txtImageName.UTF8Text = "Viz One Image"
    else '...it should be an Object Store image
\qGet the number of arrays if the picfilename divided by "\" mark
        NumberOfElements = UBound(Split(image.picFilename, "\"))
        if NumberOfElements > 0 then
            txtImageName.UTF8Text = Split(image.picFilename, "\")
(NumberOfElements)
        End if
    End if
End sub
```

Preset Image Size

A scene can be designed to use an image of a specific size. By setting a specific width and height the aspect can be controlled in accordance with the scene design. When selecting an image, a crop tool with a preset aspect ratio is automatically opened, requiring the user to crop the image. Images from Object Store will use the legacy Crop Tool, while images fetched with RestVOS from Object Store or Viz One will be cropped using Crop Service.
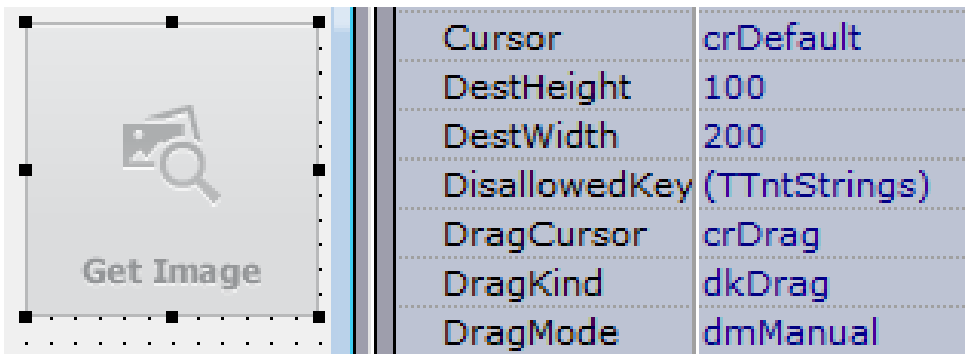
DestWidth and **DestHeight** are used as parameters to preset the crop tool and to scale the image further after save, if required.

If the property **AllowCropAspectOverride** is true, users can select a different aspect ratio than the one given by the DestHeight and DestWidth properties. This is only supported for images cropped with Crop Service.

Required Keywords

The properties RequiredKeyWords and RequiredKeywordsMode allow the Object Store (VOS), RestVOS or Viz One search to be restricted by keyword in two ways.

RequiredKeywordsMode = RequireOne:

- The media search will return images containing ONE OR MORE of the keywords specified in the RequiredKeyWords property in the template.
- When using the template in Viz Pilot News, deselecting some of the keywords is possible, and will narrow down the search. Selecting additional keywords will broaden the search.
  RequiredKeywordsMode = RequireAll:
- The media search will ONLY return images containing ALL of the keywords specified in the RequiredKeyWords property in the template.
- When using the template in Viz Pilot News, deselecting a required keyword is not possible. Selecting additional keywords will narrow down the search.

> ⚠ **Note:** The functionality of Viz One and RestVOS allow multiple keywords in the Viz One or RestVOS to map to a single keyword in the interface. If such mappings exist, any *RequireAll* searches that include such keywords will be run as *RequireOne* searches. The result is that the user will see more results than are strictly correct. However, the user will not be able to use any of the incorrect images since the "OK" button will only be enabled when an image with all the required keywords is selected. To avoid this situation, the keywords in the Viz One or RestVOS should be labeled with unique names.

Enabling the Crop Tool

If the image component uses AlwaysUseCropTool, DestHeight or DestWidth, then the crop tool will be called.

1. In Template Wizard open a template with an **image component**.
2. Select the image component and set one or both of the following properties:
   - **AlwaysUseCropTool:** Use the crop tool by default.
   - **DestHeight** and **DestWidth:** When set to values greater than 0 (zero), it will enable the crop tool. The parameters set a fixed height and width for the crop tool.

3. Select one or several image sources.
    - **ImageSources:** Valid sources for the crop tool are isImageMediaSearch, isFileOpen and isPaste.
4. **Save** the template.

Adding Person Information to the Template

In addition to fetching the person information and adding it to the proper text fields, the following procedure will use the `SC_SaveName` function to fill the Save dialog box with a save name based on the image's file name.

1. Start Template Wizard.
2. Create a new template of a scene with an image.
3. Select the image component (`TTWImageInf1`) that was added to the template.
4. Check the **Enable script** option in the Object Inspector (**CTRL + F12**) to enable scripting
5. Select the **Events** tab, and double-click the OnImageInfoChanged event to add the procedure to the Script Editor.
6. Add the following script:

```
option explicit
dim level
dim savename
function SC_SaveName
    SC_Savename = savename
end function
Sub TTWImageInf1ImageInfoChanged(ImageInfoList, PersonInfoList, KeywordsList)
    dim i
    TWUniMemo1.Lines.Clear
    TWUniMemo2.Lines.Clear
    TWUniMemo3.Lines.Clear
    for i = 0 to ImageInfoList.Count - 1
        TWUniMemo1.Lines.Add(ImageInfoList(i))
    next
    for i = 0 to PersonInfoList.Count - 1
        TWUniMemo2.Lines.Add(PersonInfoList(i))
    next
    for i = 0 to KeywordsList.Count - 1
        TWUniMemo3.Lines.Add(KeywordsList(i))
    next
    savename=ImageInfoList.Values("FileName")
End sub
```

7. **Save** the template and open it in Director or Viz Pilot News to test it.

10.9.5   World Maps Editor Component

World Maps Editor adds a map placeholder to the template.

> ⚠ **Note:** The World Maps Editor component requires a Viz World Client installation locally and a connection to one or more Viz World Servers or a Server Allocator.

The World Maps Editor component has a different workflow from an image component. In the Maps component, additional content is present in the MapXML field in the template saved on the Viz Pilot system, which makes it possible to script standard start points (for example for zooms). It also allows the correct Viz World project template to be selected, avoiding the default project error message when the maps editor opens.

> ⚠ **Note:** In Viz Pilot, it's not possible to fine tune the zoom, latitude, longitude, pan or tilt after making your map selection.

Notable Properties

- **ControlObjectName:** Links the component to an exposed scene object.
- **MapXML:** Stores the XML that is returned from the Viz World Server once the user has selected a map.
- **VizWorldEditor:** Defines the editor to be launched when a map is added to the template.
    - **vwClassic_Edition:** Classic is the default editor. All the map features are exposed and the user has full control over the map selected.
    - **vwSecond_Edition:** Second Edition is the next generation map editor, which is more user friendly. For more information on how to work with maps, please refer to the *Viz World User Guide*.

10.9.6   Panel Component

The Panel component adds a panel that can be used to group a set of components.

Notable Properties

- **Caption:** Sets a text string to identify the panel to the user.
- **Name:** Sets the name of the panel to use as a reference when scripting.
- **Height and width:** Sets the size of the panel.

Script Example (Panel and Unicode Button)

This example shows how a panel can be switched on and off using the *Visible* property. The color of the buttons change accordingly.
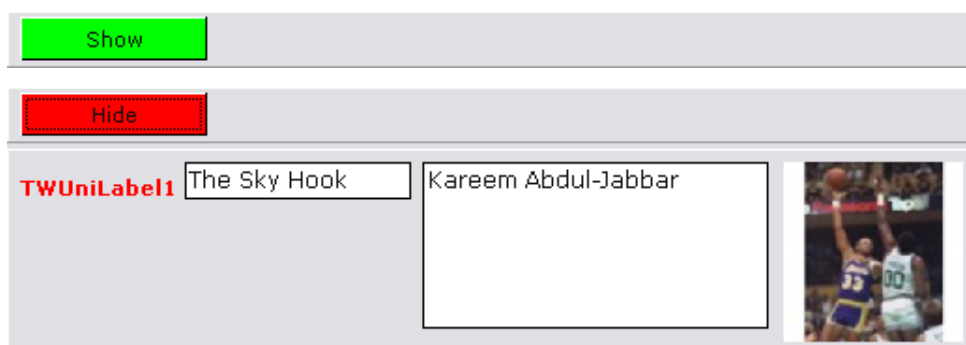
```
Sub InitForm
    TWUniLabel1.Font.Color = clRed
    TWUniButton1.Color = clLime
    TWUniButton1.Caption = "Show"
    Panel1.Visible = false
End Sub
Sub TWUniButton1Click(Sender)
    if TWUniButton1.Color = clLime then
        Panel1.Visible = true
        TWUniButton1.Color = clRed
        TWUniButton1.Caption = "Hide"
    else
        Panel1.Visible = false
        TWUniButton1.Color = clLime
        TWUniButton1.Caption = "Show"
    end if
End sub
```

✓ **Tip:** See the Speed Button Component, or the Unicode Radio Button Component, for how to create buttons that can work together as a group, and present mutually exclusive choices to the user.

10.9.7   Unicode Button Component

Unicode Button is a push button control, and it introduces several properties to control its behavior in a dialog box setting. Choose button controls to initiate actions.

Notable Properties

- **Name:** Specifies the name of the control as referenced in code.

Script Example

An example using Unicode Buttons is described in this Script Example (Panel and Unicode Button).

10.9.8   Unicode CheckBox Component

CheckBox represents a check box that can be on (checked) or off (unchecked).

Notable Properties

- **Checked:** Specifies whether the check box is checked.

Script Example

The following script example uses a sub procedure and a function. Components used are: one Unicode button, three Unicode edit fields and two Unicode check boxes.

| TWUniButton1 | TWUniEdit0 | TWUniEdit1 | ☐ TWUniCheckbox1 |
| | | TWUniEdit0 | ☑ TWUniCheckbox2 |

The Sub procedure `TWUniButton1Click` calls the function `setText` using the text string of the `TWUniEdit0` edit field as argument.

```
Sub TWUniButton1Click(Sender)
    setText(TWUniEdit0.UTF8Text)
End sub
```

The setText function tests to see if the corresponding check boxes are checked. If they are, newText is assigned to the edit fields.

```
Function setText (newText)
    If TWUniCheckbox1.Checked Then
        TWUniEdit1.Text = newText
    End If
    If TWUniCheckbox2.Checked Then
        TWUniEdit2.Text = newText
    End If
End Function
```

10.9.9   Unicode Radio Button Component

Radio buttons present a set of mutually exclusive options to the user (only one radio button in a set can be selected at a time). When the user selects a radio button, the previously selected radio button becomes unselected.

| TntGroupBox1 |
| ○ TWUniRadioButton1   ○ TWUniRadioButton2   ◉ TWUniRadioButton3 |

The Unicode GroupBox Component can be used to manage more than one set of radio buttons.

Notable Properties

- **Checked:** Determines whether the option represented by the radio button is selected.
- **Enabled:** Controls whether the radio button is enabled or disabled.

10.9.10   Unicode ComboBox Component

The ComboBox component is an edit box with a drop-down list. Users can select an item from the list or type directly into the edit box.

Notable Properties

- **UTF8Text:** Displays the current combo-box selection.
- **UTF8Items:** Lists the items (strings) in the list portion of the combo box. Use the methods of Items to add, delete and clear items, for example:

```
TWUniComboBox1.Items.Add(TWUniEdit1.Text)
TWUniComboBox1.Items.Delete(TWUniComboBox1.ItemIndex)
TWUniComboBox1.Items.Clear
```

Script Example

See the Unicode ListBox Component for a scripting example that can be used with the ComboBox.

10.9.11   Unicode ListBox Component

The ListBox displays a collection of items in a list that you can scroll. Use it to display a list of items that users can select, add, or delete.

Notable Properties

- **UTF8Items:** Lists the items (strings) in the list box. Use the methods of Items to add, delete and clear items, for example:

```
TWUniListBox1.Items.Add(TWUniEdit1.Text)
TWUniListBox1.Items.Delete(TWUniListBox1.ItemIndex)
TWUniListBox1.Items.Clear
```

Script Example

The following script example uses a sub procedure and a function. Components used are: one Unicode button, two Unicode edit fields, one Unicode check box and a Unicode list box.
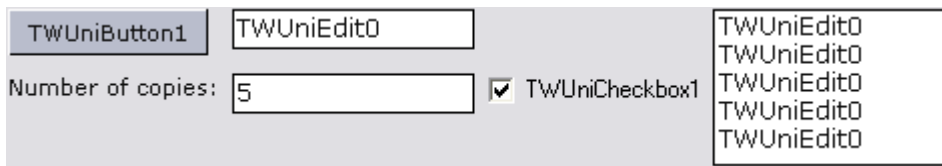
The setText function tests to see if the checkbox is checked. If checked, newText is added to the listbox the number of times given in the "Number of copies" text field (for example 5). If not checked, the text is added once to the listbox.

The Sub procedure `TWUniButton1Click` calls the function `setText` using the text string of the `TWUniEdit0` edit field as argument.

```
Function setText (newText)
    TWUniListBox1.Clear
    If TWUniCheckbox1.Checked Then
        For i = 0 To TWUniEdit1.Text -1
            TWUniListBox1.Items.Add(newText)
        Next
    Else
        TWUniListBox1.Items.Add(newText)
    End If
End Function
Sub TWUniButton1Click(Sender)
        setText (TWUniEdit0.Text)
End sub
```

✅ **Tip:** The same code can be used when working with Unicode ComboBox Component.

10.9.12   Unicode GroupBox Component

The GroupBox component represents a standard group box, used to group related controls on a form. When another control component is placed within a group box, the group box becomes the parent of that component.

See the Unicode Radio Button Component component for an illustration of GroupBox.

10.9.13   Static Image Component

A static image is often used as a background on the template or as a kind of illustration, for example a snapshot of the Viz Artist scene.

There are two ways of adding a static image to a template. One option is to add a static image component to the template and select a pre-saved image from the computer, or alternatively create a snapshot of the scene and add it to the template using Viz Engine. The snapshot is automatically appended to a static image component.

To add or edit an image, right click the static image component's anchored frame to open the context menu and click **Edit Image...**.

> ⚠️ **Note:** If an image has an alpha channel, the stretching and proportions are not respected in Template Wizard.

> ❌ **Warning:** When using the static image component, use compressed image types like JPG or lossless ones like PNG to conserve space in the Viz Pilot database. Each template and data element will save a unique copy of the image used, so keeping the image small in size is important. The image does not need to be small in terms of pixel size. The whole template is streamed to the database as a BLOB, and the image data will not be touched, so a JPG or PNG will use much less space.

10.9.14 Page Control Component

Add a multi page panel view with tabs. Select a page by clicking the page's tab that appears at the top of the control. To add a new tab sheet to a PageControl object at design time, right-click the Page Control's property page and select **New Page**. Each tab sheet that is added is given its own list of parameters in the Object inspector.
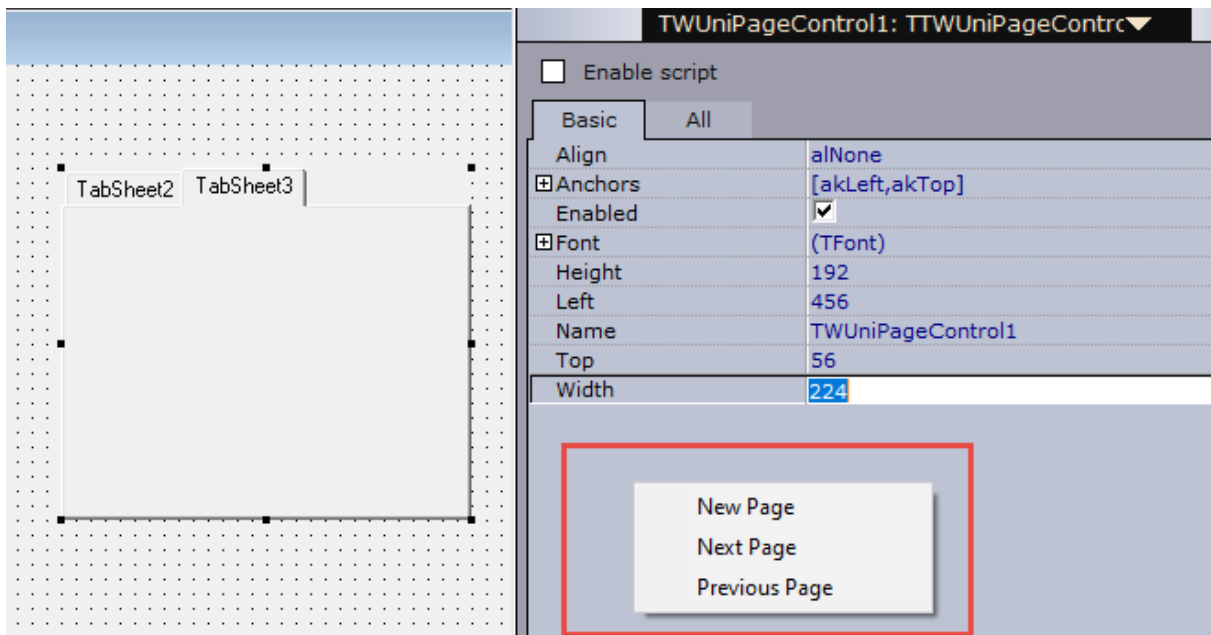
10.9.15 Template Information Component

Add the Template Information component to the template to customize the Save dialog for data elements in Viz Pilot News and Director. It can also be disabled and data element names can automatically be based on text in Unicode Edit, Unicode Memo and/or Wrap Fixed Unicode Memo components (using the UTF-8Text property). If other template information elements are required in the save name, they need to be filled

into a text component by a script. Alternatively the `SC_SaveName` function can be used, as described in Add person information to the template.

Notable Properties

- **AllowEditingOfDefaultName:** Determines if the user is permitted to edit the suggested name in the save dialog.
- **DefaultSaveName:** Set by the *SaveNameTextFields* and *Separator* properties.
- **Prefix:** Used to set a prefix for save names.
- **SaveNameTextFields:** Click the ellipsis button to open an editor to select the edit boxes that are to be used for generating a save name.
- **Separator:** Set a separator (for example a space, dot or underscore). This is used to separate text strings added to the DefaultSaveName from the edit boxes selected in the SaveNameTextFields property.
- **ShowSaveDialog:** Determines whether Director shows the save dialog when saving the template.

Advanced Properties

Some of the Template Information properties are used to set a default start time and duration for a data element when it is saved using Viz Pilot News. The properties described below are primarily used with automation systems, and can only be used with Viz Pilot News.
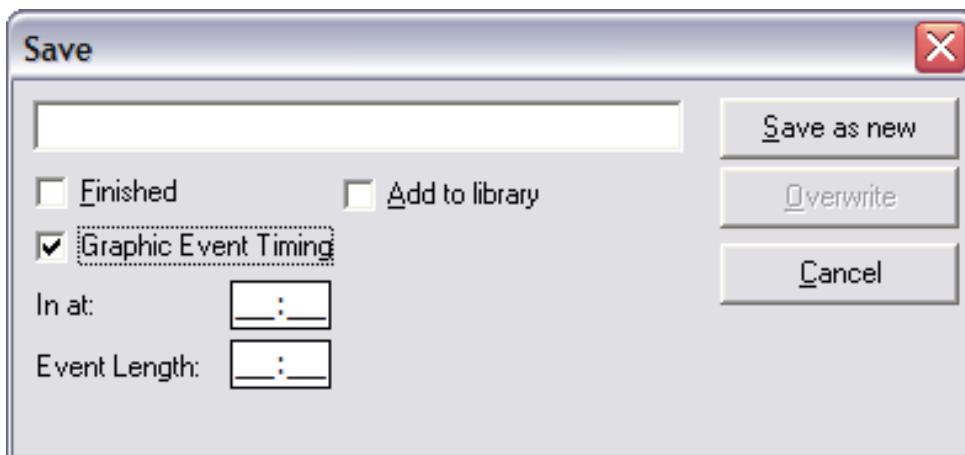
- **DefaultStartTime:** (mm:ss) Sets the start time of the data element in the playlist. The start time is relative to the start time of the playlist.
- **DefaultDuration:** (mm:ss) Sets the duration of the data element in the playlist.

> ⚠️ **Note:** *DefaultStartTime _and _DefaultDuration* always contain the current timing information for a data element. For example, a script can access these values and include timing information in the name of the data element when saving it.

- **ShowGraphicEventTiming:** Can be set as follows:
    - **stShow:** Display the *Graphic Event Timing* checkbox when Saving an Element in Viz Pilot News. The timing properties relate to *DefaultDuration* and *DefaultStartTime*.
    - **stHide:** Do not display the *Graphic Event Timing* checkbox when Saving an Element in Viz Pilot News. You can then script the values and use them as part of the template.

- **stDefault:** Use the Database Parameters (`ax_dataelement_timer_enabled`) to determine whether to show or hide the *Graphic Event Timing* checkbox.

> ⚠️ **Note:** The *stShow* and the *stHide* value will override the database setting on a template by template basis.

- **TimingMode:** Flag indicating the value of the *Graphic Event Timing* checkbox. This property only takes effect if the *Graphic Event Timing* checkbox is shown.
  - **tmManual:** *Graphic Event Timing* is not checked.
  - **tmAutomatic:** *Graphic Event Timing* is checked.

Script Example

See the Script Example (Image with Name, Unicode Edit and Template Info).

10.9.16   Person Information Component

The Person Information component connects to Object Store, and retrieves all available person information. It can be used to select and link information such as a person's TV name and title to other text fields using the predefined component properties LinkName and LinkTitle. It can also be used with scripts, or simply by replacing and reassigning the ControlObjectName of the scene property to the Person Information component.

Notable Properties

- **UTF8Text:** Displays the current drop-list selection.
- **LinkName:** Sets the TV name property to the component selected.
- **LinkTitle:** Sets the Title property to the component selected.

Using a Person Information Component in a Template
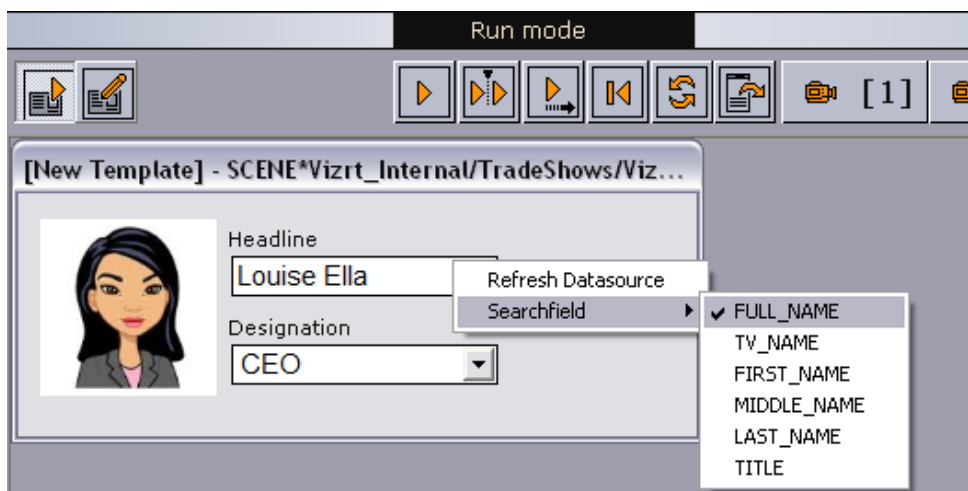
The component also contains some features that are available to the template user. A context menu appears when right-clicking the drop-down list:

- **Refresh Datasource:** Refreshes the data source, making new database entries available to the user.
- **Searchfield:** Displays a sub menu with the following options:
  - **FULL_NAME:** Displays all available full names in the drop-list.
  - **TV_NAME:** Displays all available TV names in the drop-list.
  - **FIRST_NAME:** Displays all available first names in the drop-list.
  - **MIDDLE_NAME:** Displays all available middle names in the drop-list.
  - **LAST_NAME:** Displays all available last names in the drop-list.
  - **TITLE:** Displays all available titles in the drop-list.

10.9.17   Feed Linker Component

The Feed Linker button is a push button control. When a Feed Linker button in a template is clicked, the Feed Browser window is opened and lists the items from the feed source defined in the template. For more details, see Field Linking and Feed Browsing.

When an entry is selected by the user, the data is mapped to fields in the template. This lets the user pick an item from a feed and have the corresponding data applied to fields in the template, instead of manually entering information. The template designer uses scripting to map the feed data to fields in the template.

The Feed Linker supports ATOM feeds as well as feeds with Viz Data Format (VDF) payloads, allowing data to be fed from sources such as Twitter, Flickr, Viz World Server or Pilot Data Server. Vizrt's Social TV Solution also creates feeds that can be accessed using the Field Linker.

> ✅ **Tip:** Vizrt's Social TV Solution captures content from Twitter, Facebook, Instagram, YouTube, Flickr, Picasa, Weibo and Tudou out of the box, and can be configured to capture content from almost any service that provides content in the form of XML or JSON feeds. For more information, see the *Social TV Solution User's Guide*.

This section contains the following topics:

- Notable Properties
- Events
- Using the Feed Linker Component
- Technical
- The Following Fields Can Be Linked
- The Following Fields Cannot Be Linked
- Elements in Atom Feed
- Elements in Atom Entry
- Script Example 1 - Mapping Data
- Script Example 2 - Mapping a Name
- Script Example 3 - Mapping Title, MediaContent and XML

Notable Properties

- **Source:** Specifies the URI of the feed from which the user will be able to select items.
  See also The Following Fields can be Linked and The Following Fields cannot be Linked

Events

- **OnFeedItemSelected:** Specifies the properties and methods that are available for mapping the feed data to fields in the template.
  Properties:
    - Title
    - Summary
    - ContentType
    - MediaContent(<attribute>): Returns the value of the given attribute from the main media content of the item. See Script Example 3 - Mapping Title, MediaContent and XML.
    - Xml: Returns an XML representation of the selected entry. The template designer can then extract any additional values from the item as desired by using the `entry.xml` property. See Standard Components#Script Example 3 - Mapping Title, MediaContent and XML.
    - Payload: If the item contains a payload as the main content it can be accessed using this property. The data parameter in the OnUpdatePayload method represents a payload. See Standard Components#Script Example 2 - Mapping a name.
    Methods:
    - GetLinkWithRelation. See Script Example 1 - Mapping data.

Using the Feed Linker Component

1. Add the Feed Linker component to a template.
2. In the Object Inspector set the **Source** property to the URI of a feed. For example http://site.example.com/feeds/posts.
3. See Add an event.
4. Double-click the event **OnFeedItemSelected**.
5. In the script editor, map feed data to template fields, see Script Example 1 - Mapping data.

Technical

The following XML Namespace Prefixes are used when referring to XML elements:

Prefix	URL
atom	http://www.w3.org/2005/Atom
viz	http://www.vizrt.com/types
vaext	http://www.vizrt.com/atom-ext
media	http://search.yahoo.com/mrss/
thr	http://purl.org/syndication/thread/1.0
opensearch	http://a9.com/-/spec/opensearch/1.1/

> ℹ️ **Example:** The notation `<atom:entry>` is to be interpreted as referring to the same element as
> `<entry xmlns="http://www.w3.org/2005/Atom">` .

> ⚠️ **Note:** The keywords MUST, MUST NOT, REQUIRED, SHALL, SHALL NOT, SHOULD, SHOULD NOT, RECOMMENDED, MAY, and OPTIONAL in this document are to be interpreted as described in RFC 2119.

The Following Fields Can Be Linked

- `<atom:author>`

  The personal details of the author of the selected `<atom:entry>` , which might be an individual person or an organization. The data is provided in child elements as follows:
    - `<atom:name>` : The name of the author. The value of the `<atom:name>` element inside the first `<atom:author>` element that applies for the entry is recorded as the field value. This child element is required if you are specifying the `<atom:author>` element.
    - `<atom:uri>` : A URL associated with the author, such as a blog site or a company web site. The value of the `<atom:uri>` element inside the first `<atom:author>` element that applies for the entry is recorded as the field value. This child element is optional.
    - `<atom:email>` : The e-mail address of the author. The value of the `<atom:email>` element inside the first `<atom:author>` element that applies for the entry is recorded as the field value. This child element is optional.

- `<atom:content>`

  `<atom:content>...</atom:content>`

  `<atom:content src="..." />`
    - Either inline or URL content is supported.
    - There is no type match, so it is up to the Viz Pilot operator to figure out what content can be used.
    - For URL content the resource is downloaded first and then applied as the value. If you just want the URL use `<atom:link>` .

- `<atom:entry>`

  The whole entry XML.

- `<atom:link>`

  This element defines a reference from an entry to a Web resource; in other words this is the value of the `href` attribute.

- `<vaext:locator>`

  The text of the Vizrt Atom Extension `<vaext:locator>` element which has a `vaext:type` attribute equal to the mediatype attribute of `<viz:fielddef>` element in the model.

- `<atom:published>`

  This element is a Date construct indicating the initial creation or first availability of the entry. The value of the `<atom:published>` element in the entry is recorded as the field value.

- `<atom:summary>`

  This element is a Text construct that conveys a short summary, abstract, or excerpt of an entry. The content of the `<atom:summary>` element in the entry is recorded as the field value.
- `<atom:thumbnail>`
- The value of the URL attribute of the first thumbnail element in the entry is recorded as the field value.
- `<atom:title>`

  This element is a Text construct that conveys a human-readable title for an entry or feed; the title of the selected `<atom:entry>`.
- `<atom:updated>`

  This element is a Date construct indicating the most recent instant in time when the selected `<atom:entry>` was modified. The value of the `<atom:updated>` element in the entry is recorded as the field value.

The Following Fields Cannot Be Linked

- `<atom:category>`
- `<atom:id>`
- `<atom:rights>`
- Any other elements.

Elements in Atom Feed

Elements that should be present in the `<atom:feed>` for the full user experience:

- **OpenSearch link:**

  `<atom:link rel="search" type="application/opensearchdescription+xml" href="http://example.com/opensearchdescription.xml"/>`
- Optional, needed for support for server side searching.
  - The opensearchdescription xml file must contain a template node that returns search results as an atom feed:

    `<opensearch:Url type="application/atom+xml" template="http://..." />`
- For more details see http://www.opensearch.org/Specifications/OpenSearch/1.1
- **Up link:**

  `<atom:link rel="up" type="application/atom+xml;type=feed" href="http://..." />`
- Optional, needed for nested collections.
  - The up link should link to the parent folder feed for hierarchical feeds.
  - The up link must have a type string equal to `"application/atom+xml;type=feed"`.

Elements in Atom Entry

Elements that should be present in the `<atom:entry>` for the full user experience:

- **Self link:**
  `<atom:link rel="self" type="application/atom+xml;type=entry" href="http://..." />`
- The self link must link to the URL that will return the `<atom:entry>` xml.
  - The self link must have a type value equal to `"application/atom+xml;type=entry"`.
  - Needed for refresh/update of values from the item to work.
  - Needed in combination with the up link for remembering and showing the selected feed entry in a hierarchy of feeds (folder structure). Not required for remembering selection in flat feeds since then the `atom:id` will be used.
- **Up link:**
  `<atom:link rel="up" type="application/atom+xml;type=feed" href="http://..." />`
- The up link must have a href value that is the URL of the feed the entry is in.
  - The up link must have a type value equal to `"application/atom+xml;type=feed"`.
  - Required in combination with the self link for remembering and showing the selected feed entry in a folder structure.
- **Down link:**
  `<atom:link rel="down" type="application/atom+xml;type=feed" href="http://..." />`
- Required if this `<atom:entry>` is to be considered a subfolder instead of a normal `<atom:entry>`.
  - The down link must have a type value equal to `"application/atom+xml;type=feed"`.
  - The down link must have a `href` value that is the URL of the feed (folder) the entry represents.
  - Link may contain `thr:count` attribute (RFC4685) indicating how many children there are. If the value of `thr:count` is 0 (zero) then the folder will not be loaded since it is empty. This is an optimization.
- **Thumbnail link:**
  `<media:thumbnail url="http://..."/>`
- Required for thumbnail icons to display on the entries in the feed browser.
  - The URL value must reference a JPEG or PNG image resource.
  - If many thumbnails are defined the first one will be selected as the default. This is according to Media RSS Specification Version 1.5.0.
- **Locator**
  `<vaext:locator type="application/vnd.vizrt.viz.geom">GEOM*Vizrt/Tutorials/VizPilot/...</vaext:locator>`
- Required to be able to link a tabfield to a locator. A locator is a path that is not representable with the URI `href` in an `<atom:link>`. The selected value is the text node in the locator, so it can in theory be used for any kind of text.

- For Viz Pilot, this is intended to be used to select paths to resources that are not URIs, such as resources on the Graphic Hub that are currently in use.
- The `type` must match the tabfield type.
- The value from the first locator that matches will be used.
- If no locators matches an error message will be logged in Viz Pilot.
- The Viz Engine types are:
  - TEXT: `text/plain`
  - RICHTEXT: `application/vnd.vizrt.viz.richtext+xml`
  - MATERIAL: `application/vnd.vizrt.viz.material`
  - DUPLET: `application/vnd.vizrt.viz.duplet`
  - TRIPLET: `application/vnd.vizrt.viz.triplet`
  - FONT: `application/vnd.vizrt.viz.font`
  - CLOCK: `application/vnd.vizrt.viz.clockcommands`
  - GEOM: `application/vnd.vizrt.viz.geom`
  - IMAGE: `application/vnd.vizrt.viz.image`
  - AUDIO: `application/vnd.vizrt.viz.audio`
  - VIDEO: `application/vnd.vizrt.viz.video`
  - MAP: `application/vnd.vizrt.curious.map`
  - TRIOSCROLL: `application/vnd.vizrt.trio.scrollelements+xml`
- **Other links:**
  ```
  <link rel="..." href="http://..." type="..."/>
  ```
- Required in order to select the URL to external resources. These can be images, videos, text, and so on.
  - Limitation 1: The atom 1.0 specs allow multiple links with the same relation, but when linking a tabfield to a link in Viz Pilot, only the first link will be selected.
  - Limitation 2: Viz Pilot can only use the `href` URL as the value. It cannot download the resource at the URL and use its contents as the value. The Viz Engine control plugin for the property has to be able to understand the URL and download the resource.

Script Example 1 - Mapping Data

Link data (Title, Summary, ContentType) from a news feed to editbox fields in a template, including getting the link of the feed item itself using GetLinkWithRelation.

```
Sub TWFeedLinker1FeedItemSelected(entry)
    SetUnicodeValue TTWUniEdit1, entry.Title
    SetUnicodeValue TTWUniEdit2, entry.Summary
    SetUnicodeValue TTWUniEdit3, entry.ContentType
    SetUnicodeValue TTWUniEdit4, entry.GetLinkWithRelation("self")
End sub
```

Script Example 2 - Mapping a Name

Link a person's TVname in a person feed to an editbox in the template.


```
Sub TWFeedLinker1FeedItemSelected(entry)
    set payload = entry.Payload
    tvnameedit.text = payload.Item("tvname").Value
    titleedit.text = payload.Item("title").Value
    firstnameedit.text = payload.Item("firstname").Value
    middlenameedit.text = payload.Item("middlename").Value
    lastnameedit.text = payload.Item("lastname").Value
End sub
```

> ⚠ **Note:** Although the TTWUniDbPersonComboBox component can also be used to map person name, it only supports data from the Object Store. The Feed Linker component supports all asset search providers (for example RestVOS and Viz One).

Script Example 3 - Mapping Title, MediaContent and XML

Link text, images and XML from feed data to a template.
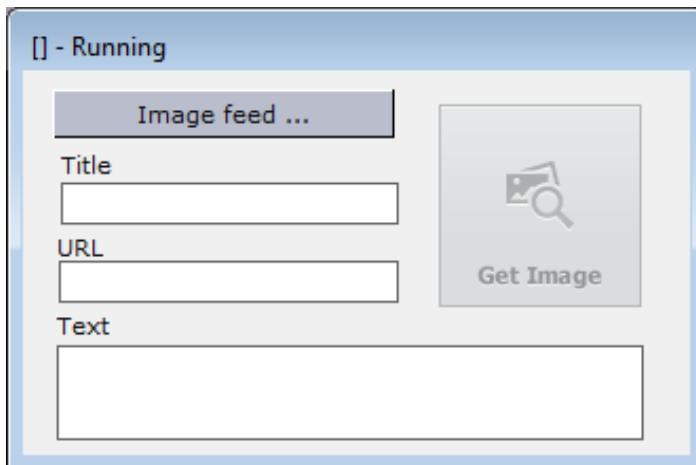

```
Sub TWFeedLinker1FeedItemSelected(entry)
    TTWUniEdit1.Text = entry.Title
    set media = entry.MediaContent
    if not isnull(media) then
        TWImageInf1.picFilename = media.Url
        TWUniEdit1.Text = media.Url
        TWImageInf1.UpdateThumbnail
    end if
    TWUniMemo1.Text = entry.xml
End sub
```

> ⚠ **Note:** If images are linked in using the method in this example script, then all Viz Engines must have internet access. See the *Viz Artist* and *Viz Engine* guides for more information on supported image formats.

10.10   Database Components

Database components provide the means to connect to any database that can be used with the Open Database Connectivity (ODBC) API.

There are two ways of setting up database communication: either by using the ActiveX Data Objects (ADO) components or by writing scripts without using database component. Writing scripts is often the preferred solution as this lets users find more elaborate and complete script examples on the Internet than what is covered in the current user manual.

> ⚠️ **Note:** Many examples on the Internet use the **Server.** notation since many scripts are used for accessing a web server. This is not required with Template Wizard.

The most common ADO components available are ADO Connection Component, ADO Query Component, ADO Command Component, ADO Table Component, and ADO DataSet Component. Not all components are required at once to establish a database connection and to retrieve or store data. To understand their purpose and area of use, please read the following subsections.

> ⚠️ **Note:** An ADO provider represents one of a number of types of access, from native OLE DB drivers to ODBC drivers. These drivers must be installed on the client computer. OLE DB drivers for various database systems are supplied by the database vendor or by a third-party.

This section contains the following topics:

- Setting Up a Basic Database Connection
- ADO Connection Component
- ADO Query Component
- ADO Command Component
- ADO Table Component
- ADO DataSet Component

10.10.1   Setting Up a Basic Database Connection

Two ways of setting up a basic database connection are described below:

- Example I - Using scripting
- Example II - Using components
  Both scripting examples use a standard Microsoft Access database. The database contains a table (dbTable) that consists of the following fields; id, headline and source.

The template itself has three labels and listboxes corresponding to the database table fields.

Example I - Using Scripting

In this example, database components are not used, as using them would not show all the possibilities of scripting.

ID	Headline	Source
5	Viz|Template Wizard ADO scripting example	Vizrt

1. Initialize variables and values. Option Explicit requires that all variable names be defined (with the Dim statement). Const is similar to a variable, except that its value cannot be changed at runtime:

```
Option Explicit
Const adCmdText = 1
Const adStateOpen = 1
Dim cnVizrt
Dim cmdQuery
Dim rsVizrtStory
```

2. Create a database connection using the Connection object:

```
Set cnVizrt = CreateObject("ADODB.Connection")
```

3. A minimum set of connection properties can be set. In this case the Provider and Data Source. In this case User ID and Password is also provided. If none is required the sentence can be removed:

```
cnVizrt.ConnectionString = _
    "Provider=Microsoft.Jet.OLEDB.4.0; " & _
    "User ID= SomeUser; Password; SomePassword; " & _
    "Data Source=C:\test.mdb"
```

4. Open the database:

```
cnVizrt.Open
```

5. Create a query string using the Command object:

```
Set cmdQuery = CreateObject("ADODB.Command")
With cmdQuery
    Set .ActiveConnection = cnVizrt
    .CommandText = "SELECT id, headline, source " & _
    "FROM dbTable " & _
    "WHERE source = 'Vizrt'"
    .CommandType = adCmdText
    Set rsVizrtStory = .Execute
End With
```

6. Test to see if the database connection was successful:

```
If rsVizrtStory.State = adStateOpen Then
    MsgBox("Recordset opened successfully.")
End If
```

7. Clear redundant data from the listboxes:

```
TWUniListBox1.Clear
TWUniListBox2.Clear
TWUniListBox3.Clear
```

8. Add data from the database to text fields:

```
Do While Not rsVizrtStory.EOF
    TWUniListBox1.Items.Add(rsVizrtStory.Fields("id"))
    TWUniListBox2.Items.Add(rsVizrtStory.Fields("headline"))
    TWUniListBox3.Items.Add(rsVizrtStory.Fields("source"))
    rsVizrtStory.MoveNext
Loop
```

9. Delete all objects from memory that are no longer required:

```
rsVizrtStory.Close
Set rsVizrtStory = Nothing
Set cmdQuery = Nothing
cnVizrt.Close
Set cnVizrt = Nothing
```

Example II - Using Components

Setting up a database connection is also possible using database components. A simple setup can be described as follows:

1. Add an ADO Connection component to the template and set the following properties:
   - **ConnectionString**: Information required to connect the ADO Query component;

   ```
   Provider=Microsoft.Jet.OLEDB.4.0; Data Source=C:\test.mdb;
   ```

   - **Provider**: This property is auto generated based on the connection string property.
2. Add an ADO Query component to the template and set the following properties:
   - **Connection**: Select the ADO Connection name from the drop-list.
   - **SQL**: Click the browse button and enter the query string. For example:

   ```
   SELECT id, headline, source FROM dbTable WHERE source = 'Vizrt';
   ```

3. Check the **Connected** property in the Object Inspector for the ADO Connection component. A Database login window appears asking for User Name and Password. If one is not set, click **OK**.
4. Check the **Active** property in the Object Inspector for the ADO Query component.

5. Open the ADO Query context menu by right clicking the component icon on the template editor. Click the Fields Editor ... menu option.

6. Select fields by clicking the **arrow and plus** button in the Fields Editor window. If the connection is not active, a dialog appears asking for User Name and Password.

7. Add Unicode labels and listboxes to the template.
8. The following script will list the result from the ADO Query component in the listboxes:

```
Option Explicit
Dim index
ADOConnection1.Connected = True
ADOQuery1.Active = True
```

9. Test to see if the database connection was successful:

```
If ADOQuery1.State = 1 Then
    MsgBox("Recordset opened successfully.")
    MsgBox(ADOQuery1.RecordCount)
End If
```

10. Test to see how many records were found based on the SQL statement:

```
MsgBox(ADOQuery1.RecordCount)
```

11. Clear listboxes:

```
TWUniListBox1.Clear
TWUniListBox2.Clear
TWUniListBox3.Clear
```

12. Before iterating a record set, it's recommended to reset the cursor in order to start from the top when running the script several times:

```
ADOQuery1.First
```

13. The **For** statement iterates according to the RecordCount -1 adding data to the listboxes. Since the loop starts at 0, the RecordCount method must be set to -1. This prevents the loop from giving a copy of the last record:

```
For index = 0 To ADOQuery1.RecordCount −1
    TWUniListBox1.UTF8Items.Add(ADOQuery1id1.Text)
    TWUniListBox2.UTF8Items.Add(ADOQuery1headline1.Text)
    TWUniListBox3.UTF8Items.Add(ADOQuery1source1.Text)
    ADOQuery1.Next
Next
```

⚠ **Note:** The field names (for example `ADOQuery1headline1`) are auto-generated names. All database fields, selected from the Fields Editor, are added as template components.

14. Disconnect the database connection and deactivate the query component:

```
ADOConnection1.Connected = False
ADOQuery1.Active = False
```

Although the ADO Query component can establish a connection directly using its ConnectionString property, it is often desirable to use an ADO Connection to share a single connection among several ADO components. This can reduce resource consumption, and allows transactions to be created that span multiple data sets.

10.10.2   ADO Connection Component

The ADO Connection component connects to an ADO data store. It encapsulates the ADO connection object. The connection provided by a single ADO Connection component can be shared by multiple ADO command and data set components through their Connection properties. This can reduce resource usage, and allows for transactions to be created that span multiple data sets.

ADO Connection allows for control of the attributes and conditions of a connection to a data store. Use the ADO Connection properties to control attributes such as record locking scheme (optimistic versus pessimistic), cursor type, cursor location, isolation level, and connection time-out.

Notable Properties

- **Connected:** Specifies whether the connection is active.
- **ConnectionString:** Specifies the connection information for the data store.
- **DefaultDatabase:** Indicates the database the ADO connection uses by default.
- **LoginPrompt:** Specifies whether a login dialog appears immediately before opening a new connection.

- **Provider:** Specifies the provider for the ADO connection. Entering a valid ConnectionString usually generates this input.

10.10.3   ADO Query Component

ADO Query provides the means for issuing SQL against an ADO data store. As ADO Query is a query-type data set it can encapsulate an SQL statement, enabling applications to access the resulting records.

Notable Properties

- **Active:** When checked, sets the query component active. If the SQL query fails it will not be set to active.
- **Connection:** Use Connection to specify an ADO connection component to connect to a data store.
- **SQL:** SQL query parameter.

10.10.4   ADO Command Component

ADO Command represents an ADO command object. Use ADO Command for issuing commands against a data store accessed through an ADO provider.

The ADO Command component executes the command specified in its CommandText property. One command may be executed at a time. If the command includes Parameters, they are specified in its Parameters property. The command is executed by a call to the Execute method.

ADO Command can either use a ADO Connection object to connect to a data store (through its Connection property) or connect directly to the data store if the connection information is specified in the ConnectionString property.

ADO Command is most often used for executing data definition language (DDL) SQL commands, or to execute a stored procedure that does not return a result set. For SQL statements that return a result set, ADO DataSet, or ADO Query is better suited. However, the Execute method of ADO Command is capable of returning a record set. However, a separate ADO data set component is required to use that record set.

10.10.5   ADO Table Component

ADO Table is a table-type data set that represents all of the rows and columns of a single database table. The component encapsulates a table accessed through an ADO data store. It can access data in a single database table using ADO. It provides direct access to every record and field in an underlying database table, but can also work with a subset of records within a database table using ranges and filters.

Notable Properties

- **Active:** When selected, the component is active.

- **Connection:** Specifies the ADO connection component to use.
- **Name:** Specifies the name of the control as referenced in code.
- **ReadOnly:** When checked ,the database connection is set to be read only.
- **RecordCount:** Indicates the total number of records in the record set.
- **Recordset:** Provides direct access to the ADO record set object.

10.10.6   ADO DataSet Component

ADO DataSet is a generic ADO data set control, and can be used in place of ADO Query. It enables data representation from one or more tables in a database and allows data-aware components to manipulate data by connecting with a DataSource (non-Unicode) component. The data source component is the link between the visual components displaying the data and the database.

Unlike the other ADO data set components, ADO DataSet is not a table-type, query-type, or stored procedure-type data set. Instead, it can function as any of these types.

As a table-type data set (ADO Table), ADO DataSet is able to represent rows and columns of a single database table. To use it in this way, set the CommandType property to *cmdTable* and the CommandText property to the name of the table.

As a query-type data set (ADO Query), ADO DataSet can specify a single SQL command that is executed when a data set is opened. To use it in this way, set the CommandType property to *cmdText* and the CommandText property to the SQL command to be executed.

Notable Properties

- **Active:** When checked, it sets the query component active. If the SQL query fails, it is not possible to set it active.
- **Connection:** Use Connection to specify an ADO Connection object to use to connect to a data store.
- **ConnectionString:** Set ConnectionString to specify the information needed to connect the ADO Connection component to the data store.
- **DataSource:** Links the edit control to the data set that contains the field it represents.

11 Template Manager

Template Wizard's template manager tool organizes templates and their scenes into concepts, templates and variants of templates.

The interface consists of three panes for browsing, and a fourth pane that is exposed when managing the concepts, concept templates and variants of templates.

The **Concepts - ALL**, **Concept templates** and **Template variants** panes are used for creating concepts, adding templates to a concept and defining variants of a template within a concept, respectively. It's also possible to edit and delete existing concepts, and their templates and variants.
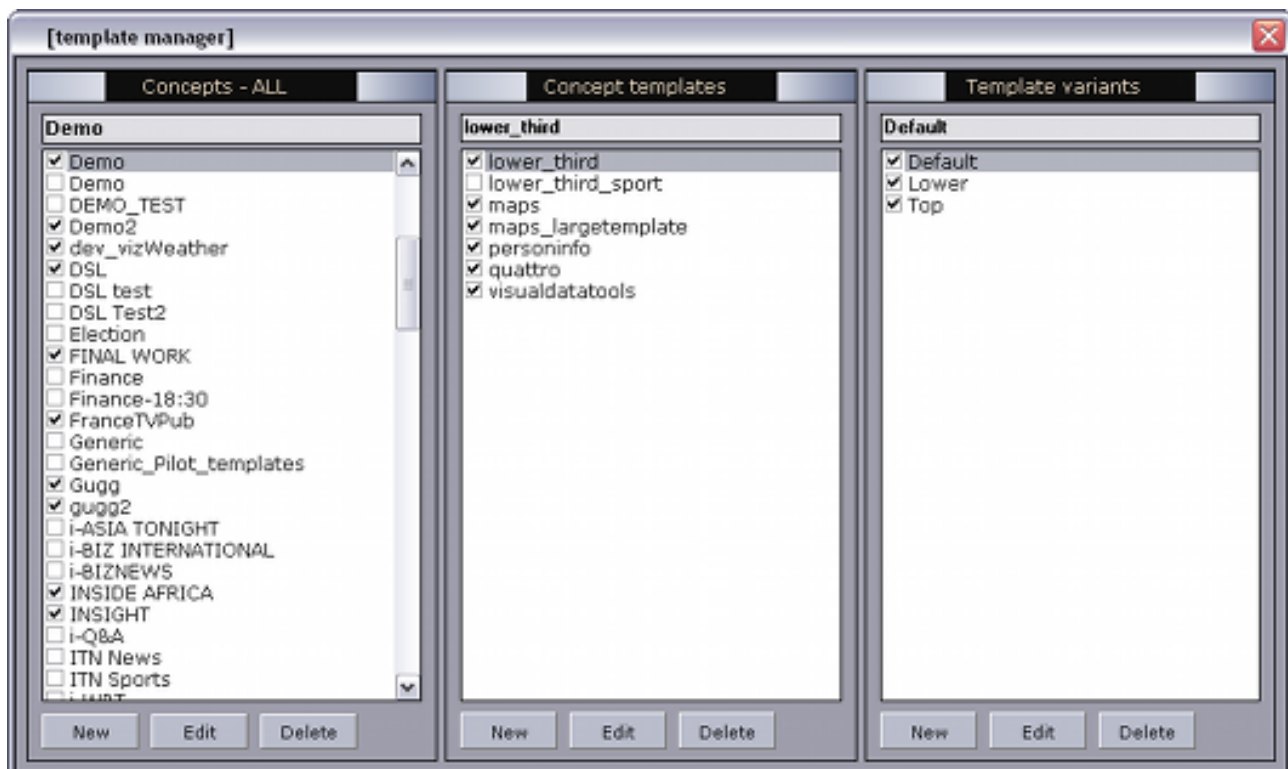
This section contains the following topics:

- Concepts
- Variants
- VCP-Templates

11.1 Concepts

11.1.1 Why Concepts are Used

When working with concepts there are a few things to keep in mind. First of all, a concept is a placeholder for a set of templates, like a folder. Secondly, concepts are about reuse of data.

A news concept for the afternoon show might look different than the concept for the evening news; however, if you rerun a story from the afternoon show you most probably would like to reuse the data in your evening show. Switching concepts and thereby the look and feel of your graphics, but reusing the data makes a lot of sense. Another use case is to output the same news story to multiple channels at the same time (see Forked Execution).

To achieve this you need to create your concepts in Template Manager and add all relevant templates to it. Once your first concept is finished you simply duplicate the concept for your next concept, swap the scenes used by the templates in the concept, and you are done.

This workflow allows you to build a single set of templates for multiple concepts. However, in order for your templates to be reused this also requires the graphics designer to create an identical number of scenes with identical properties exposed for each concept. Although scenes may therefore differ in terms of design, their properties do not differ (the exposed properties' name, type, and order remain the same). For example, if a one line lower third is created for the afternoon show, there must be a one line lower third for the evening show with the exact same set of properties exposed in order for your template to work.

11.1.2 Working with Concepts

The Concepts - ALL pane lists all available concepts stored on the database. Checked concepts are visible to users. It is also possible to create, update and delete concepts using Template Manager.

Concept Settings

Selecting a concept from the Concepts -ALL pane will automatically display the templates that belong to the selected concept in the Concept templates pane.

If you are creating concepts for broadcast and online publishing you should consider applying a naming convention that will make a distinct separation between the two concepts. This will help users avoid those concepts that are clearly marked for a different use case then their own. Most often graphics will be different in terms of animation and aspect and will not necessarily work in both situations. For more information, see Variants.

Also note that concept names created for online publishing must match the concept names defined in the online publishing system.

Concept Settings Table

Name	Description
Name of concept	Sets the name of the new concept.
Id	Is automatically set when the concept is saved.

Name	Description
Active	Toggles (active/inactive) the visible state of the concept template in the Viz Pilot system's applications. This can also be done by checking or unchecking the checkbox beside the concept listed in the Concepts - ALL pane.

Creating a New Concept

1. Click the **New** button in the **Concepts - ALL** pane to open the **New concept** editor.
2. Enter a name for the concept in the **Name of concept** field.
3. Click **Save**.

Editing a Concept

1. Select a concept and click the **Edit** button in the **Concepts - ALL** pane to open the **Edit concept** editor.
2. Edit the Concept settings and click **Save**.

Deleting a Concept

- Select a concept and click the **Delete** button in the **Concepts - ALL** pane and in the confirmation dialog box click **Yes** to confirm the deletion.

> ❌ **Caution:** All associated data with the concept will be permanently deleted from the database.

11.2   Variants

When a concept's template is selected, the template's variants are displayed in the Template variants pane, where you can add, edit and delete variants controlling standard scenes.

11.2.1   Variant View

When a concept's template is selected, the template's variants are displayed in the Template variants pane, where you can add, edit and delete variants controlling standard scenes.

Transition logic variants can only be edited for setting an external ID, description, category assignment and stop points. Transition logic does not support variants since transition logic is meant to provide the variants all in one with seamless transitions between them. Hence, transition logic templates will only have a Default template variant.

Variants of a template used for online publishing must be saved with variant names (`<width>x<height>`) according to the resolution of the image required by the publishing template (for example 285x170, 620x350, 720x295 and so on). The variant must also be saved to a concept that is used for online publishing (see Concepts).

Variants for standard scenes are a set of similar scenes that can be controlled using the same template. For this to work, all scenes must be built the same way. The scenes must have the same path to the objects that are controlled by the template, or alternatively they can be set up with equal control channels. Scene objects that are controlled by a template must be of the same kind for the template to work across different scenes.

For example: A template controlling a text object with Field Identifier=1 in one scene cannot automatically control an image object with Field Identifier=1 in another scene.

Template Variant Settings

Name	Description
Description of variant	Sets a description for the variant. The name will show up as a list item in the Template variant list.
Template ID	Displays the ID of the selected template.
Active	Toggles (active/inactive) the visible state of the template variant in the Viz Pilot system's applications. This can also be done by checking or unchecking the checkbox next to the template variants listed in the Template variants pane.
Scene folder	Sets the path to the folder that contains the scenes for the template variant. If a template's scene folder is changed, the scenes for every variant of that template (in a specific concept) will be re-imported from Viz.
Scene name	Lists all scenes available in the Scene folder.
Stop points (Default)	Sets the number of stop points in the scene. This is done to show the operator during playout how many times Continue has to be executed when the variant is played. It is also used by automation systems to enable automatic execution of the Continue commands.
Custom ID (Optional)	Assigns a custom ID to a variant. This is only needed if some external logic is to be executed with the variant or a selection of variant with the same ID are to be extracted and processed.
External ID	Assigns a unique ID used for integration with external systems such as a newsroom system. The List button opens a window that shows the external IDs in use. Right-click an ID to edit the variant with that ID. See also **External ID** used with Import Components.
Category	Sets the category for which the template variant can be filtered. See Categories and Channels.

11.2.2 Working with Variants

Adding or Editing a Variant

1. Click the **New** or **Edit** button in the **Template variants** pane to open the **New variant** or **Edit variant** editor.
2. Enter a name in the **Description of variant** field.

3. Select a scene from the **Scene name** drop-list.
4. *Optional:* Enter an **External ID**.
5. *Optional*: Set a **Category**.
6. Click **Save**.

> ⚠️ **Note:** It's not possible to create variants for transition logic based templates.

Deleting a Variant

- Click the **Delete** button in the Template variants pane, and confirm the deletion.

> ❌ **Caution:** Deleting a variant deletes all data elements that reference it.

11.3   VCP-Templates

Templates are fill-in forms for your graphics. In addition, they can be reused for variations of the graphics within the same concept and copied to other design Concepts for reuse. All use cases support reuse of data, meaning that the user only needs to fill in data once.

This section contains the following topics:

- Templates View
- Working with Templates

11.3.1   Templates View

Templates are listed in the Concept templates list. Checked templates are visible to users. To view a concept's templates, simply select a concept from the Concepts list. Templates can be added, copied, edited or deleted from a concept.

Selecting a template will automatically list the template's variants in the Template variants pane.

Concept Template Settings

Name	Description
Template Name	Sets the selected template to be added to the concept.
Template ID	Displays the ID of the selected template.
Active	Toggles (active/inactive) the visible state of the concept template in the Viz Pilot system's applications. This can also be done by checking or unchecking the checkbox next to the concept template listed in the Concept templates pane.
Scene folder	Sets the path to the folder that contains the scenes for the template. If a template's scene folder is changed, the scenes for every variant of that template (in a specific concept) will be re-imported from Viz.
Thumbnail	Placeholder for the scene's thumbnail.

11.3.2   Working with Templates

Adding a Template
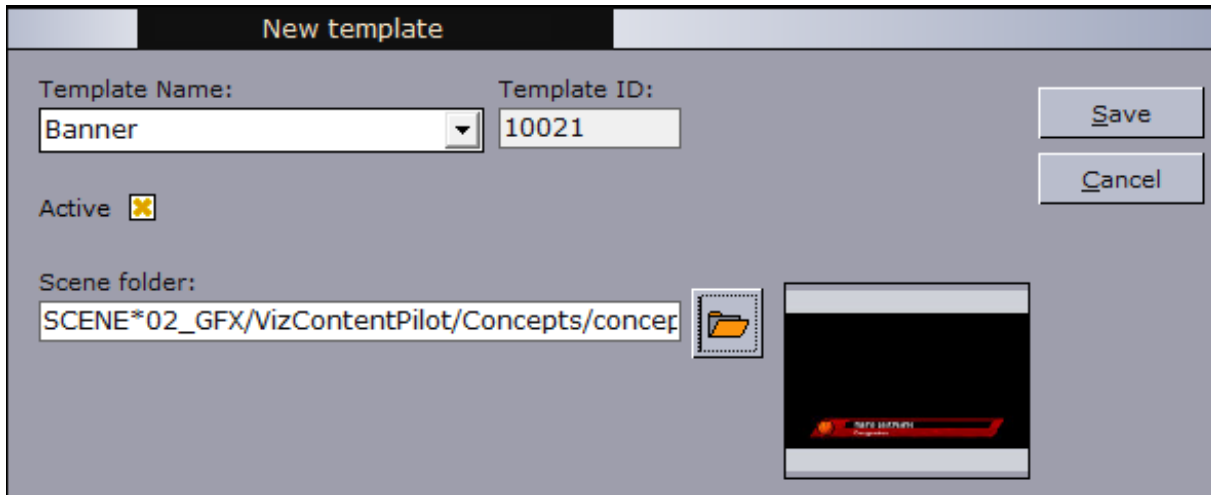
1. Select a concept from the **Concepts - ALL** pane in which to create the template.
2. Click **New** in the **Concept templates** pane.
3. Select a template from the **Template Name** drop-list.
4. **Browse** the Viz directory for the scene to use with the template, and click **OK**.
5. Click **Save**.

> ⚠️ **Note:** The procedure differs somewhat for transition logic concept templates. See Adding a Transition Logic Template below.

Adding a Transition Logic Template

1. Select a concept from the **Concepts - ALL** pane to copy the template to.
2. Click **New** in the **Concept templates** pane.
3. Select a transition logic template from the **Template Name** drop-list.
4. **Browse** the Viz directory for the transition logic scenes to use with the template. Make sure the selected scenes are tagged OK, and click **OK**.
5. Click **Save**.

Copying an Unassigned Template from the Resource Panel

1. In the Resource Panel select the **<Unassigned>** concept in the template view.
2. In Template Manager select the desired concept from the **Concepts - ALL** pane.
3. Drag and drop the template from the Resource Panel to the **Concept templates** pane in Template Manager.
4. Click **Yes** to confirm the action.

Copying a Template from One Concept to Another

1. Select a concept to copy a template from.
2. From the **Concept templates** pane, drag and drop the template onto one of the concepts in the **Concepts - ALL** pane.

> ❌ **IMPORTANT!** This type of copying does not result in two independent templates. The new template will have the same Template ID as the original, and changes made to either template will be reflected in both.

> ⚠ **Note:** The template cannot be copied if the template exists within the other concept.

Copying a Transition Logic Template from One Concept to Another

1. Select the template from the **Concept templates** pane that will be copied to the new concept.
2. Drag and drop the template onto the designated concept listed in the **Concepts - ALL** pane.
3. Confirm that the template should be copied to the new concept.
4. **Browse** the Viz directory for the transition logic scenes to use with the template. Make sure the selected scenes are tagged OK, and click **OK**.
5. Click **Save**.

> ⚠ **Note:** Scenes used by the originating template cannot be selected.

Editing a Template

1. Select a template from the **Concept templates** pane, and click the **Edit** button to open the **Edit template** editor.
2. Set new properties for the **Active** and **Scene folder** settings.
    - If you have copied a template from one concept to another you can use the **Scene folder** to set the path for your new design concept.
3. Click **Save**.

Deleting a Template

> ❌ **Caution:** Deleting a template deletes all data elements that reference it.

1. Select a template and click the **Delete** button located in the **Concepts template** pane.
2. Click **Yes** in the confirmation dialog box.

12 Crop Service And Crop Tool

Crop Tool and Crop Service are tools that allow for basic cutting, zooming and rotating of images. They provide an efficient way for journalists and editors to quickly add new images to a data element.

Crop Service makes it possible to crop RestVOS images and images from any Viz One server. Crop Service will be automatically used for all images except for those from the original Object Store, which will still be handled by Crop Tool.

Crop Tool allows images to be cropped from Object Store. Crop Tool is legacy and may be deprecated in future versions of Viz Pilot. Object Store and Crop Tool are not supported when using a Graphic Hub Database.

> ✅ **Tip:** Using RestVOS and Crop Service is the recommended workflow combination.

This section contains the following topics:

- Features of Crop Service and Crop Tool
- Using Crop Service
- Using Crop Tool
- Setting up Crop Service and Crop Tool

12.1   Features Of Crop Service And Crop Tool

12.1.1   Comparison of Crop Tool and Crop Service

| Feature	Crop Tool	Crop Service
Supports images from Viz One	No	Yes
Supports images from RestVOS (see the RestVOS section in the Pilot Data Server Administrator Guide)	No	Yes
Supports images from (legacy) Object Store	Yes	No
Images are saved as PNG	Yes	Yes
Will re-size the image to template and image properties	Yes	Yes (*)
Will keep the aspect of the cropped area	Yes	Yes
Needs a shared drive/UNC path for storing stills	Yes	No
Can paste images directly from the clipboard into a template	Yes	No
Can rotate source images 90 degrees in both directions	Yes	No
Black will be used as filling to be able to use the whole image width or height	No	No
Re-crop of images is possible (as image source is linked in)	No	No

> ⚠ **Note:** (*) Where destHeight and destWidth are defined in the template then, in most cases, Crop Service resizes the image to match these properties. If the selection is smaller than destHeight or destWidth then a warning is shown but the selection may still be used. Note that if destWidth*destHeight < 10,000 pixels 2, then the image is not resized, but the selection may still be used.

All edited images from both Crop Tool and Crop Service are saved as Portable Network Graphics (PNG) images, and saved with the same image and person information as the original image, if that exists. The new file is given a default prefix *crop_paste*<random characters>. Images with alpha or key values are preserved.

Crop Service caches cropped images on file on the Pilot Data Server, as well as on the Viz Engine.

Crop Tool saves the cropped image to the VOS-file-share, and from there it is handled the same as any other VOS image.

12.2   Using Crop Service
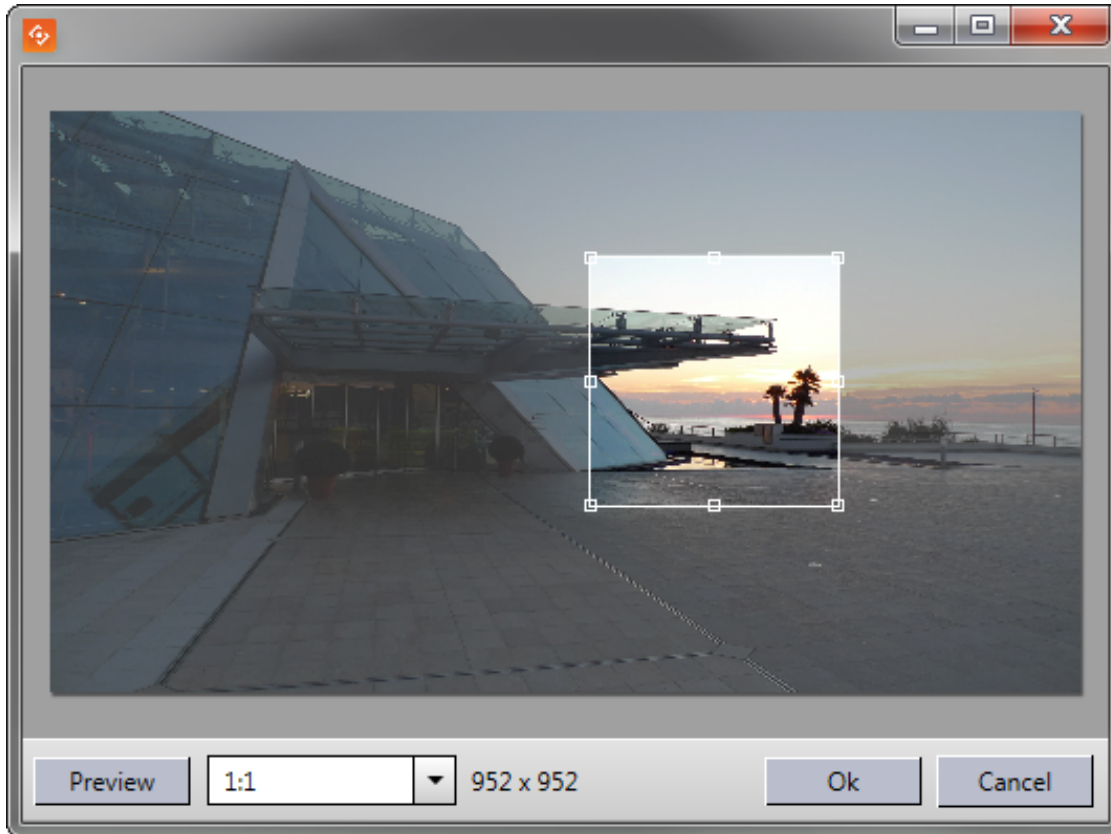
12.2.1   Crop Service Functions

Menu Bar

- **Preview:** Opens a preview window to display the selected crop.
- **Target Ratio:** Select a ratio from the drop-down list and then select a crop in the edit area. (Manual ratio, Original ratio, 4:3, 16:9, 1:1).
- **Selection Size:** Displays the pixel dimensions of the selected crop.
- **OK:** Use the selected crop.
- **Cancel:** Resets editing and closes the Crop Service window.

Edit Area

- **Drag** the anchored frame to move the selected crop area.
- **Resize** the anchored frame to change the selected crop area.

12.2.2   Cropping an Image with Crop Service

1. If **AlwaysUseCropTool** or **DestHeight** and **DestWidth** are set in the template, Crop Service will open automatically when an image is dragged to the image placeholder.
2. Select the desired area.
    - *Optional*: Click **Preview** to preview the selection.
3. Click **OK** to confirm the selection.

> ✅ **Tip:** The maximum size of images created by Crop Service can be configured. Any crop that is bigger than this limit will be resized, respecting the aspect ratio of the crop. If the DestHeigth or DestWidth properties on the image component are used, then the smaller constraint will be used (see Crop and Scale Image). Maximum size applies even if no cropping is done. For more information, see the Setting Parameters using Pilot Data Server section in the Pilot Data Server Administrator Guide or how to set Viz Pilot Database Parameters.

12.3   Using Crop Tool
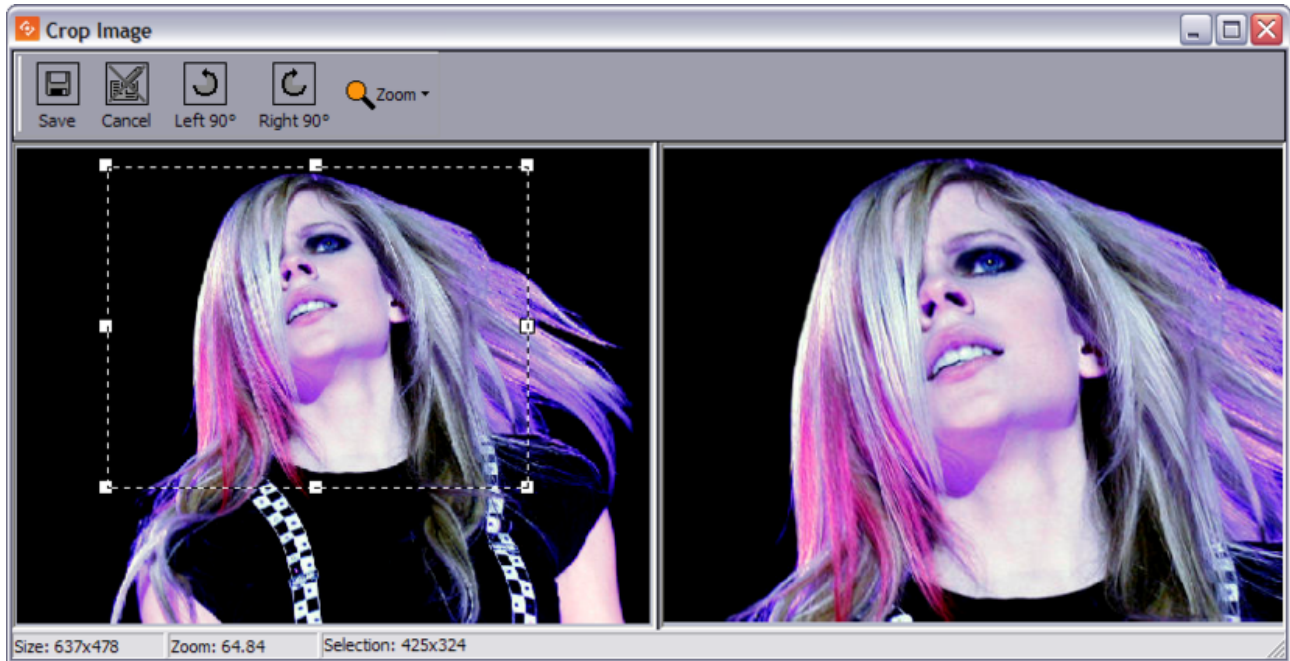
This section contains the following topics:

- Images
- Crop Tool Functions
- Cropping an Image with Crop Tool
- Using an Image from the Clipboard
- Images Edited with Crop Tool
- Scene and Template Design

12.3.1   Images

Depending on the design of the template, Crop Tool can use images from Object Store, any local computer file location, or the Windows Clipboard.

> ⚠ **Note:** A cropped image can only be used once - a user cannot re-crop or search for an already cropped image. To create a new cropped image, the user must find the original image and add it to the Windows Clipboard again.

Crop tool is also used in Object Store; however, images are not saved to the Object Store database when used in Viz Pilot News. Even if an image from Object Store is used, the original image in Object Store will therefore not be affected.

> ⚠ **Note:** Crop Tool images differ from Object Store and Viz One images in that it is not possible to fully control what images are used. Policies must therefore be clear on such issues as using copyrighted images.

12.3.2   Crop Tool Functions

Crop Tool is divided into three areas: a menu bar, an edit area (left), and a preview area (right).


Menu Bar

- **Save:** Saves the image to the image directory set in the crop tool save path.
- **Cancel:** Resets editing and closes the Crop Tool window.
- **Left 90:** Rotates the image 90 degrees to the left.
- **Right 90:** Rotates the image 90 degrees to the right.
- **Zoom:** Zoom the editable version of the image to a size of; 500%, 200%, 150%, 100%, 75%, 50%, 25% and 10% of the original image size. In addition there is an option to Fit the image to the available space. Zoom can also be performed by using the mouse wheel. Zoom does not resize the saved image.

12.3.3   Cropping an Image with Crop Tool

1. Open a template or a data element.
2. Click the image placeholder.
3. Search for a stored image.
4. Drag the image to the placeholder.
5. Adjust the anchored frame and/or rotation of the image, and click **Save** (**CTRL + S**) on the tool bar.

12.3.4   Using an Image from the Clipboard
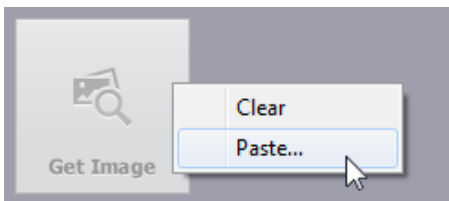
⚠️ **Note:** This procedure requires that the template's image component uses the ImageSources.isPaste parameter.

1. Open a template or a data element.
2. Search for and copy an image (for example web-based systems or file shares).
3. Right-click the image icon and select **Paste**.
4. Adjust the anchored frame and/or rotation of the image, and click **Save** (**CTRL + S**) on the tool bar.

12.3.5   Images Edited with Crop Tool

All **edited images** are saved as Portable Network Graphics (**PNG**), which is a bit-mapped image format that employs lossless data compression.

An edited image is saved with the same image and person information as the original image. The new file is given a default **prefix** "*Crop* ". The image Description field is also updated by adding the following text: *"_Crop of: Description"* where the phrase *Description* represents the image description from the original image. Images with key are also preserved.

12.3.6   Scene and Template Design

Calling Crop Tool from a Template

A scene can be designed to use an image of a specific aspect ratio. This is done in Template Wizard, by using the Image Component and its parameters *DestWidth* and *DestHeight* to set the width and height. If an image is selected that does not match the parameters, Crop Tool will be automatically opened, forcing the user to edit the image according to the pre-configured parameters.

Crop and Scale Image

If the crop uses the DestWidth/Height *ratio*, the cropped image will be rescaled to fit the DestWidth/Height *values*.

Example - Setting Image Size

For example, if the template designer has set the *DestWidth* and *DestHeight* to 200 x 100 pixels (aspect 2:1) to **set the aspect ratio and image size**, and we have an original image of 2000 x 1500 pixels (4:3), then:

- If the template user selects an area of 1250 x 625 pixels (2:1), the image will be cropped and *scaled down* to 200 x 100 pixels.
- If the template user selects an area of 200 x 100 pixels (2:1), the image will be cropped to 200 x 100 pixels.
- If the template user selects an area of 100 x 50 pixels (2:1), the image will be cropped and *scaled up* to 200 x 100 pixels.

> ⚠️ **Note:** Although a warning appears if an image is scaled up, the image can still be used. The image may appear pixelated on air.

Example - Setting Aspect Ratio

If the *DestWidth* x _DestHeight _result in a value less than 10,000, then the Crop Tool will use the aspect ratio of the width/height values, but the image will not be resized.

For example, you may set the *DestWidth* and *DestHeight* to 16 and 9 to **force an aspect ratio** of 16:9. Instead of resizing the cropped image to a size of 16 x 9 pixels, Crop Tool will use the size of the selection. For example, if the original image is 2000 x 1500 pixels, then:

- If you select an area of 800 x 450 pixels (16:9), the image will be cropped to 800 x 450 pixels. No resizing will occur.

See Also

- General Preferences preferences - Crop Tool Save Path
- Working with Stillstore Images
- Object Store Crop Tool

12.4 Setting Up Crop Service And Crop Tool

Some configuration is required for Crop Tool and Crop Service.

12.4.1 Configuring Crop Service

> ⚠ **Note:** Before configuring Crop Service, a Pilot Data Server must be installed, configured and running (one per separate schema used in the Viz Pilot Database). See Install the Data Server and Install the Crop Service.

- If Crop Service and Pilot Data Server are installed on different machines, see Configure the connection from Crop Service to Pilot Data Server below.
- If performing Viz One searches from Viz Pilot News, see configure search providers (Viz One).
- If performing RestVOS (see the Pilot Data Server Administrator Guide) searches from Viz Pilot News.
- If configuring the Viz Pilot system to use Crop Service (see **Set the VCP Parameters for Crop Service**), see the **Setting Parameter** section in the Pilot Data Server Administrator Guide.

> ⚠ **Note:** In order for Crop Service to support .TGA files, an appropriate Windows Imaging Component (WIC) codec must be installed.

12.4.2 Configuring the Crop Service to Pilot Data Server Connection

> ⚠ **Note:** The steps below are not required if Crop Service is installed on the same machine as Pilot Data Server, since Crop Service is properly configured by default.

If Crop Service is installed on a *different* machine than the Pilot Data Server, configure the connection to Pilot Data Server as follows:

1. In Windows, open the Services dialogue and stop Crop Service.
2. Open the Crop Service configuration file. This must be done "As Administrator".
   For example, run Notepad (as administrator), and in Notepad open the config file:
   `C:\Program Files\Vizrt\Crop Service\CropServiceHost.exe.config`
3. In the config file, find the following section:

```
<userSettings>
    <CropServiceHost.Properties.Settings>
        <setting name="PilotDataServerUri" serializeAs="String">
            <value/>
        </setting>
    </CropServiceHost.Properties.Settings>
</userSettings>
```

4. Replace `<value/>` with

   `<value>http://YOUR\-PDS\-HOST\-HERE:8177</value>`
   and save your changes.
5. Start Crop Service.
6. Verify that the configuration is correct by viewing the log file at `C:`

   `\ProgramData\Vizrt\Logs\CropService\cropservice.log`
   If you see an error similar to:
   ERROR: Failed to get payload from: http://localhost:8177/vcp/parameters, Error: Unable to connect to the remote server the configuration is *incorrect*, and you must repeat steps 1-6.

12.4.3   Configuring Crop Tool

- Set the desired Crop Tool Save Path in General Preferences.

12.4.4   Enabling Image Cropping in a Template (Crop Tool and Crop Service)

1. Start Template Wizard.
2. Open a template with an **image component**.
3. Select the image component and set one or both of the following properties:
   - **AlwaysUseCropTool**: Enables the user to always use crop tool/crop service when an image is added to the template.
   - **DestHeight** and **DestWidth**: When set to values greater than 0 (zero), it will enable crop tool/crop service. The parameters set a fixed height and width for the cropped image. Using this setting will force the user to use the configured height, width, and aspect. If the selection is greater than the actual height and width parameters, the image will be scaled to fit, allowing the user to select a greater area, and conversely.
4. Select one or more image sources:
   - **ImageSources**: Valid sources for crop tool are isImageMediaSearch, isFileOpen and isPaste.
5. **Save** the template.

> ⚠️ **Note:** isPaste only works with Crop Tool, see Configuring Crop Tool.

> ⚠️ **Note:** In order to use an image from the clipboard with Viz One or RestVOS, and Crop Service, the image must first be ingested into Viz One or RestVOS and then used in Crop Service.

13   Object Store
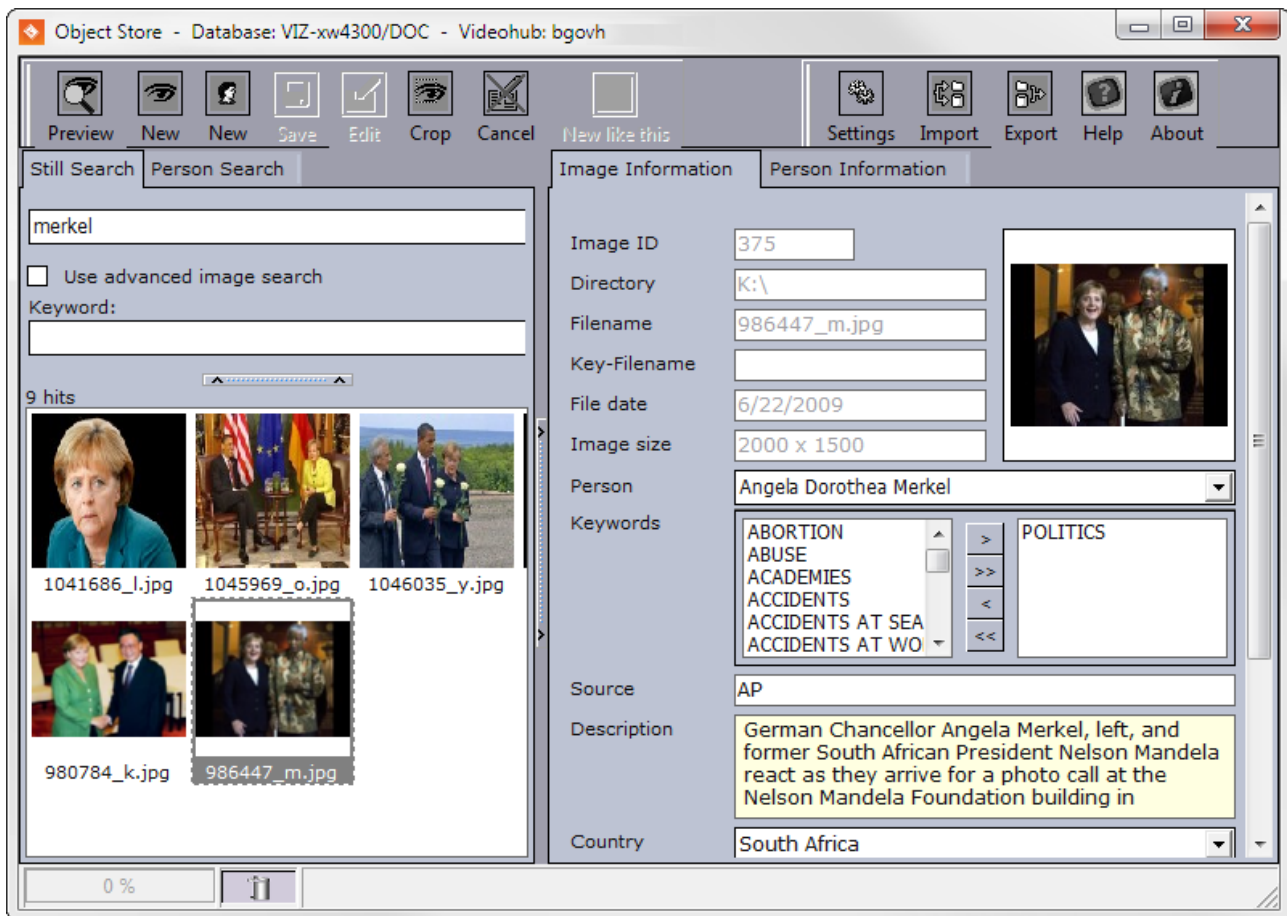
Object Store is a tool to add, edit and search for still images (stills), person information and videos from a database. By letting you organize your own media library by adding custom meta data fields, Object Store also helps you customize your workflow, and helps you create your stories quickly and efficiently.

> ⚠️ **Note:** Object Store was previously known as Viz Object Store, so the acronym VOS is still in use in some places.

> ⚠️ **Note:** Object Store is not supported when using a Graphic Hub Database.

This section contains the following topics:

- Getting Started with Object Store
- Object Store Workflow
- Object Store Configuration
- Object Store Toolbar
- Object Store Search Options
- Object Store Status Bar
- Object Store Crop Tool
- Object Store Export and Import

- Object Store Settings
- Object Store Information Panels

13.1   Getting Started With Object Store

Object Store is installed using the Viz Pilot system installer (see Viz Pilot System Installation). Object Store is dependent on a running Viz Pilot database, an image file share and/or a Viz One for access to content.

13.1.1   Starting

Object Store is a standalone application that can be started from the program menu or a desktop shortcut.

1. Open the program from the Windows Start Menu (**Object Store**), or the desktop icon.
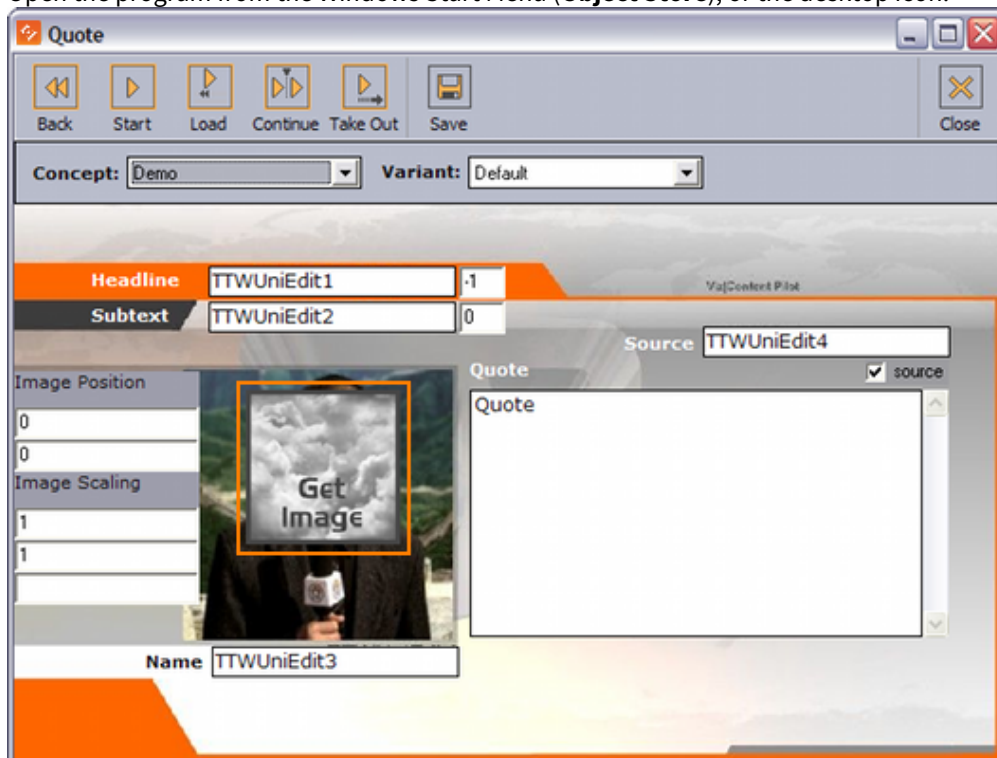
Object Store is also accessible through Director, Viz Trio, Template Wizard, Viz Pilot News and others. The integrations only allow for searching the Viz Pilot database (in addition to Viz One).

13.2 Object Store Workflow

The basic workflow from graphics design to template and content creation is as follows:

- A graphics scene is created in Viz Artist.
- A template, based on a scene, is created in Template Wizard.
- A story is typically created in a newsroom system where a template is used to open Object Store for adding stills, person information, audio and video. The story is placed in a playlist which in turn is read and played out by a control application such as Director or Viz Trio.

13.3   Object Store Configuration

Configuration parameters are set after installation (see Initialization and Configuration Files and Save Paths).

13.3.1   Object Store Initialization File

Object Store depends on an initialization file to read and connect to its database. On installation, Object Store will use the ContentPilot.ini. However, Object Store also supports using `objectstore.ini`. If you use a separate initialization file for Object Store you only need to add the database settings.

> ❌ **IMPORTANT!** If Director is started with the `-inifile` command line option, then values in `objectstore.ini` are overridden by `ContentPilot.ini`. Director will not read Object Store connection data from `objectstore.ini`.

13.3.2   Parameters

- **login:** User ID for the database connection.
- **pwd:** Password for the database connection.
- **utf8:** Sets whether the program and database should use UTF8 font encoding.
- **nls_lang:** If UTF-8 is used for the database, the NLS_LANG environment setting must be here.
- **name:** This entry must contain the connection string for the database in the form: "hostname/instance name". If a database client (for example Oracle Client) is used, enter the TNS name.

13.3.3   Configuration

Object Store configuration is simple. Using the Initialization File Editor, make sure that the *Database Parameters* in the Object Store Initialization File are in accordance with your preferences. Next, configure the Save Paths.

See Also

- Configuration
- Initialization and Configuration Files

13.4   Object Store Toolbar

- **Preview Image:** Previews the image shown in the Image Information panel in the Image Preview window.
- **New Image (CTRL + N):** Opens a file browser and a new Image Information panel for adding a new image to the database.
- **New Person:** Opens a new Person Information panel for adding a new person to the database.
- **Save (CTRL + S):** Saves Image Information or Person Information to the database and shared file server.
- **Edit (CTRL + E):** Edits the selected still image or person for editing Image Information or Person Information, respectively.
- **Crop:** Opens the image shown in the Image Information panel in the Object Store Crop Tool.
- **Cancel:** Cancels New or Edit operations for Image Information or Person Information.
- **New Like This:** Adds a new image by using the same metadata as that of an image already opened. You can choose to delete or keep the old image. The following metadata is reused: Person information, Keywords, Source, Description, Country and City.
- **Settings:** Opens the Object Store Settings window.
- **Import:** Opens the Object Store Import window for browsing and selecting exported files.
- **Export:** Opens the Object Store Export tab for exporting images.
- **Help:** Opens the *Viz Pilot User Guide* (this document).
- **About:** Displays the Object Store version information.

13.5   Object Store Search Options
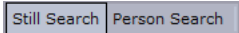
The search panel consists of two tabs: Still Search and Person Search. The tabs can be enabled or disabled through the Object Store Settings window.

Still searches are used for image and image information searches. Person search is used for person information and related image and image information searches. Both searches will search for related information on the Viz Pilot database.

When searching for an image or person, it is possible to double-click a search result item to preview and view the metadata in the corresponding information tab. Any linked image and person information is also available by selecting the respective information tab.

The Context Menus available in the results pane can be used to sort, filter and delete images and person information.

13.5.1   Still Search

Still searches are performed in two ways. A standard search consists of a search criteria and an optional selected keyword from the **Keyword** drop-list, while an advanced search can refine the search by narrowing the hits down to certain dates, multiple keywords, location and custom made additional fields.

Standard Search
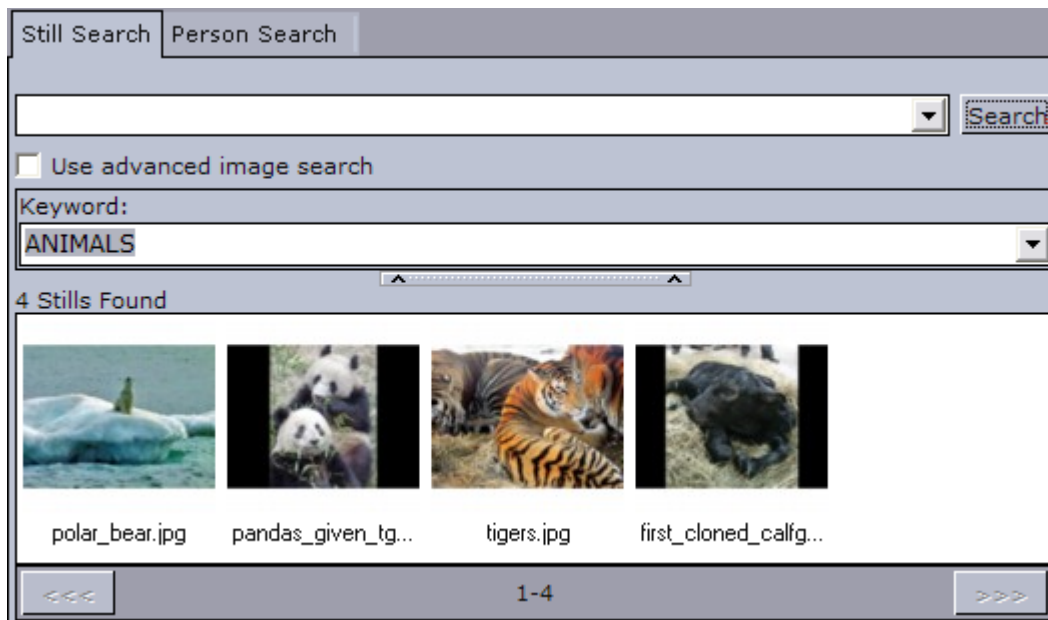
The standard still search contains a combo box, a check box, and a drop-list. The first search field searches the image Description, Country and City and the person's first and last name. The **Keyword** drop-list sets a selected keyword that searches the Image information Keywords field.

> ✅ **Tip:** Right-click to open the context menu and sort the search results.

Advanced Search

As with Standard search, Advanced search is capable of doing simple searches. In addition, searches can be refined by using more than one search criteria.

- **Keyword:** Use multiple keywords for a more precise search with an *unlimited* number of keywords selected.
- **Country and City:** Country and City corresponds to the drop-down list and text field used when registering Image information.
- **Dates:** Clicking on the arrow opens a calendar where dates can be selected. There are two check boxes for dates for searches based on dates in three different ways.

- If only the first check box is selected, a search will be made for images from the selected date to the present date.
- If only the second check box is selected, a search will be made for images from before the selected date.
- By selecting both check boxes, a search will be made for images within the selected dates.
- **Must have all keywords:** Select this option to show hits that contain all the selected keywords.
- **Portraits only:** Set the search to only search for images set as portraits linked to a person (see Person Information).
- **Additional fields:** Adds customized additional fields with the option of adding a search criteria. Clicking **Add** adds the customized search to the standard search field.

- The additional fields search criteria entered in the example above is appended to the existing search criteria.

Stills with Key

al_gore_right_... barack_obama... john_sidney_m...
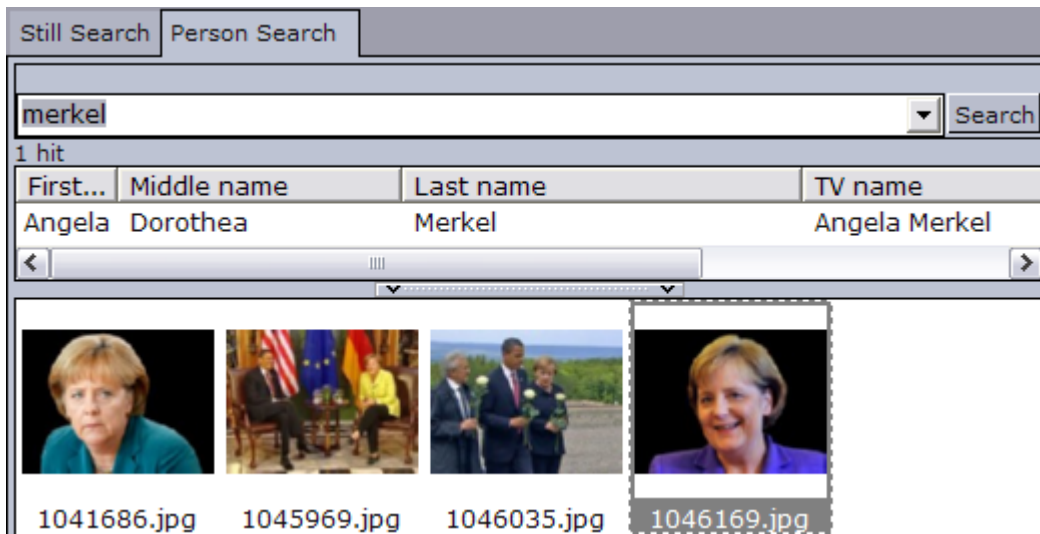
Still images with a key/alpha channel will be tagged with a **K** in the upper right corner.

13.5.2   Person Search

The following fields in the Person information panel are used in Person search: **First name**, **Middle name** and **Last name.**

Searching for a person works the same way as a Standard Search. The only difference is that results are presented as a list of persons that match the search criteria.

When clicking a list item, all images linked to that person are shown. Double-clicking a list item opens the Person Information panel. Double-clicking any of the images opens the Image Information panel related to the image.

13.5.3   Context Menus

All search options have context menus with options for sorting, deleting person or image, and restoring deleted images from the bin.

Sorting
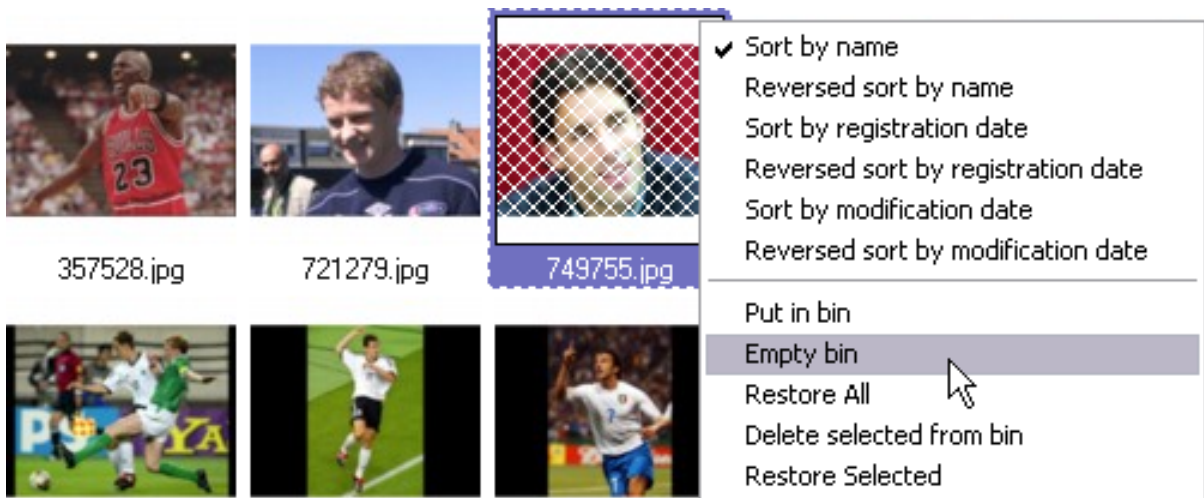
- **Sort by name:** Sorts by name in ascending alphabetical order.
- **Reversed sort by name:** Sorts by name in descending alphabetical order.
- **Sort by registration date:** Sorts by registration date in ascending order.
- **Reversed sort by registration date:** Sorts by registration date in descending order.

- **Sort by modification date:** Sorts by modification date in ascending order.
- **Reversed sort by modification date:** Sorts by modification date in descending order.

> ⚠️ **Note:** Search result are ordered in ascending (a,b,c) or descending (c,b,a) alphabetical order.

Delete, Purge and Restore

> ❌ **Caution:** Deleting person information is permanent. Images linked to a person are permanently deleted. It is not possible to restore images and information.

Person Search:

- **Delete person(s):** Permanently deletes persons from the database. See Delete person information.
- **Still Search:** Searches for still images.
- **Put in bin:** Puts selected stills into the bin.
- **Empty bin:** Purges all previously deleted stills from the database and shared file server.
- **Restore All:** Restores all stills from the bin.
- **Delete selected from bin:** Purges selected stills from the database and shared file server.
- **Restore Selected:** Restores selected stills from the bin.

> ⚠️ **Note:** If a password is set, enter a valid password to confirm the deletion of images.

> ⚠️ **Note:** The delete and restore options cannot be used with Viz One instances.
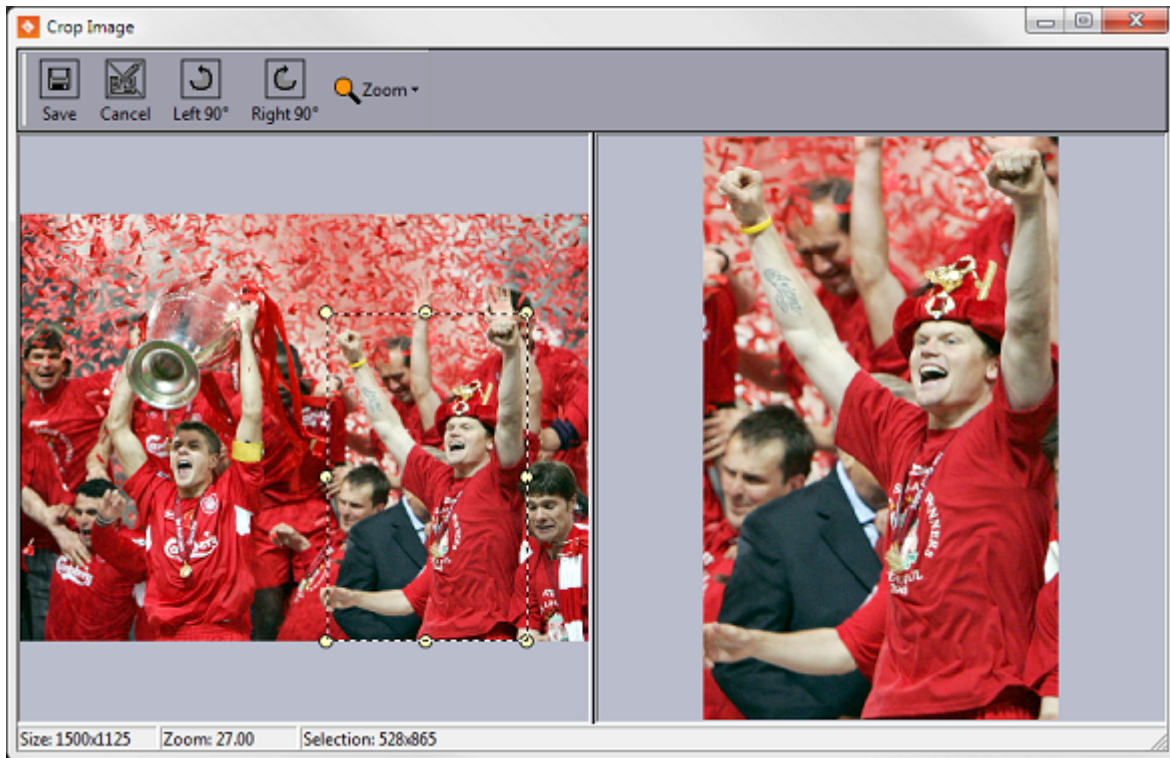
See Also

- Image Information

13.6 Object Store Status Bar

The status bar consists of two parts: a **search progress bar** and a **bin**.

To use the bin, simply drag and drop an image onto the bin icon.

13.7 Object Store Crop Tool

Crop Tool is used for basic cutting, zooming and rotating of images. An edited image does not affect an image that has already been saved.

For information about using Crop Tool, see Crop Service and Crop Tool.

13.8   Object Store Export And Import

This section describes how to export images and import images. To enable the Export/Import tab, click the **Import** or **Export** button on the Object Store Toolbar. Once enabled it will remain visible until Object Store is restarted.

Object Store uses an Oracle database and a shared file server to store image and person information. Original images are stored on the shared file server. Information and a thumbnail of the original image are stored on the database.

13.8.1   Object Store Export
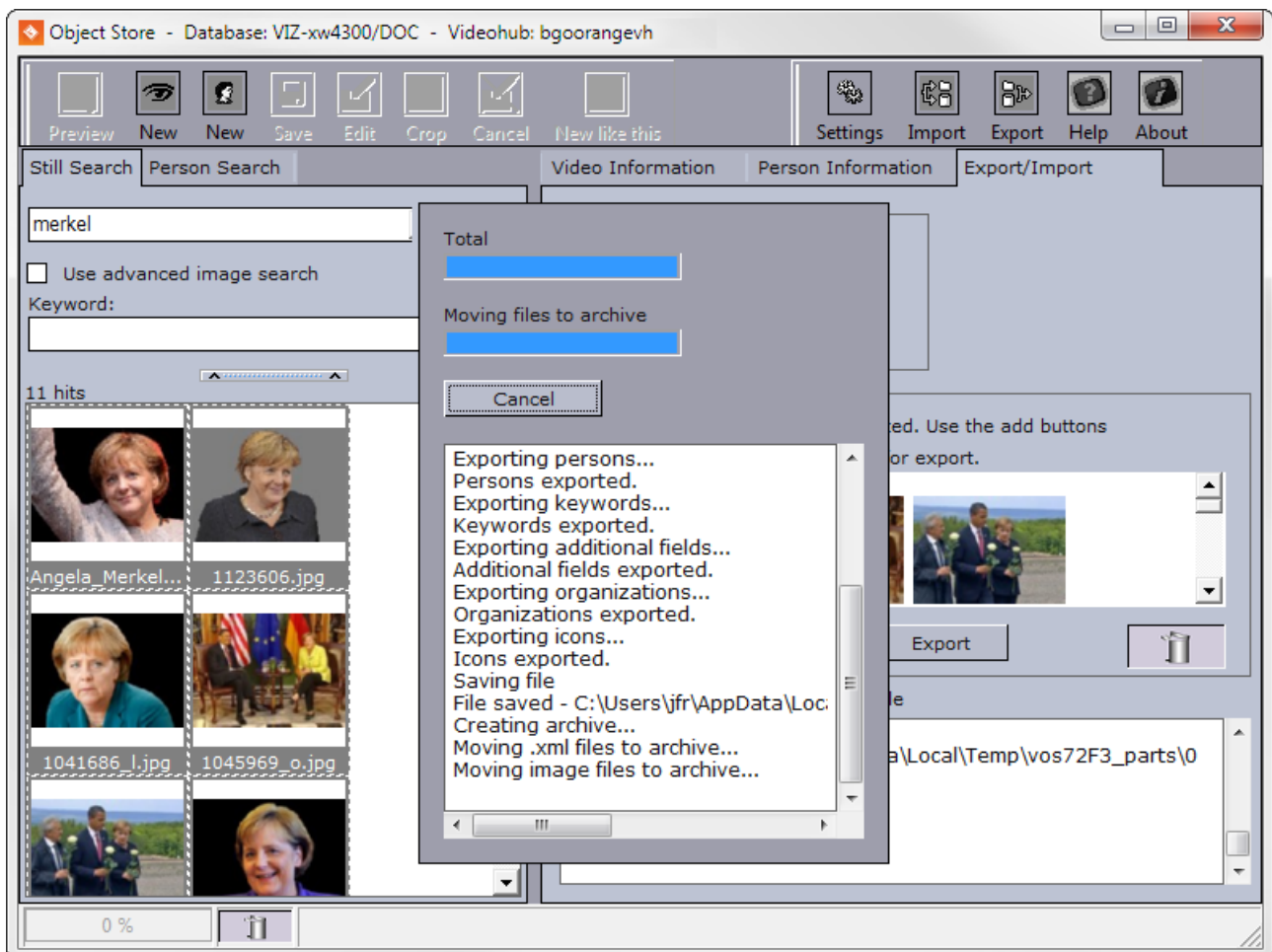
Access the image export option by clicking the **Export** button on the toolbar. This can be used to export images and related information to other Object Store installations, for example.

The Export buttons for both export options create a compressed file with original images, thumbnails and image data. The compressed file can be opened with a traditional extraction tool, or using Object Store.

The lower right section of Object Store's main window shows step-by-step status when importing or exporting images.

- **Export**: Exports all or selected images to a compressed file (for example *foo.vos*).

- **Add All**: Adds all images based on the search result.
- **Clear**: Removes all images from the drop-zone.
- **Bin**: Removes selected images from the drop-zone using drag and drop.

Exporting Selected Images

1. Click the **Export** button on the main menu to open the **Export/Import** tab.
2. Search for the images to be exported.
3. Drag and drop pictures onto the drop-zone in the **Select pictures for export** frame.
4. Click the **Export** button.
5. Enter a file name and click **Save**.

Exporting All Images

1. Click the **Export** button on the main menu to open the **Export/Import** tab.
2. Click the **Export** button in the **Direct Export** frame.
3. Enter a file name and click **Save**.

13.8.2   Object Store Import

Import is used when moving or copying images and information from one system to another. While importing the images, Object Store will extract all thumbnails and image data from the attached XML file, store it in the Viz Pilot database and place the original images to the location specified in the Object Store settings.

> ⚠️ **Note:** Duplicates are not imported.

Importing Exported Images

1. Click the **Import** button in Object Store and select the compressed file (for example *foo.vos*), or
2. Double-click the compressed file to open Object Store, or
3. Right-click and select **Import** to open Object Store.

Extracting Exported Images

- Open the compressed file (for example *foo.vos*) using an extraction tool (for example 7-zip).

13.9   Object Store Settings

To access the **Settings** window, click the **Settings** button on the Object Store Toolbar.

The Settings window consists of six tabs:

- General
- Keywords
- Additional Fields
- Save Paths
- Organizations
- Countries

13.9.1   General

The General settings tab contains the Object Store environment variables. For example, the ability to enable and disable different search options.

To access the **General** tab, click the **Settings** button on the Object Store Toolbar.
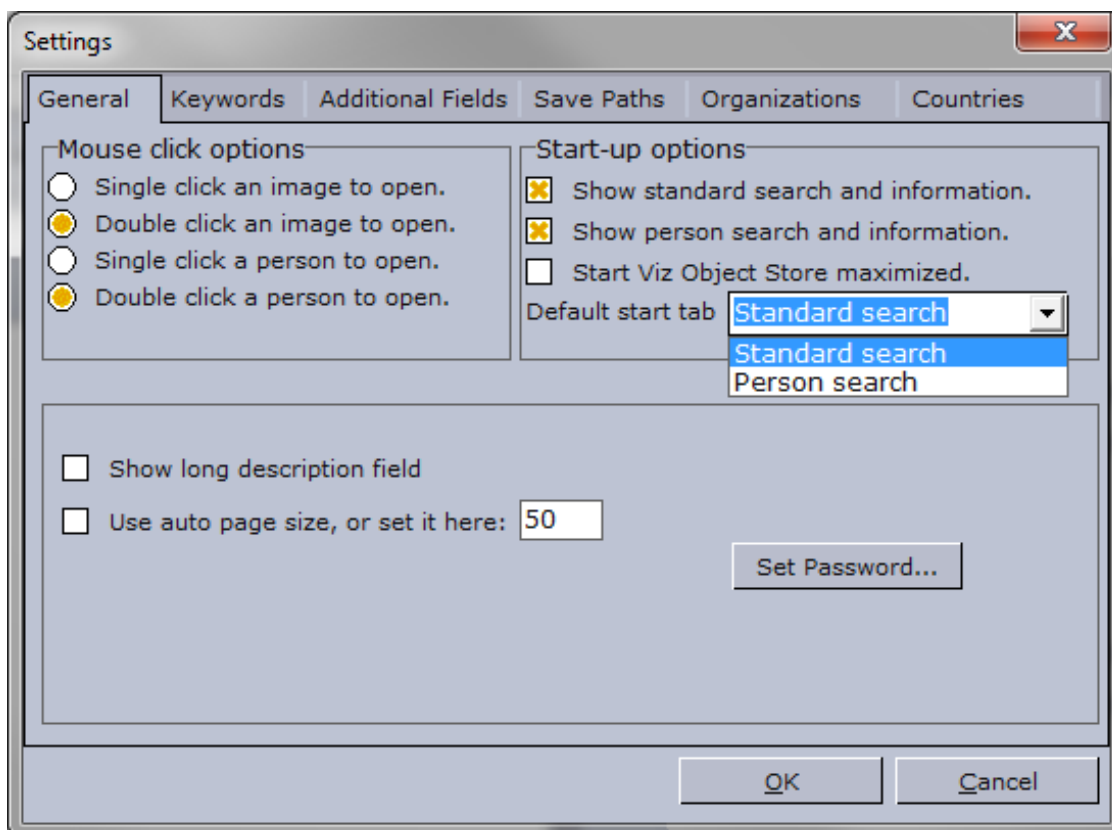
13.9.2   General Settings

- **Mouse click options:** Select to use single or double-clicks when opening an image or person in the information panels. By default, these are set to double-click.
- **Start-up options:** Switch on or off search and information options, Start Object Store maximized and select a Default start tab. The options are:
    - Show standard search and information.
    - Show person search and information.
    - Start Object Store maximized.
    - Default start tab: Sets one of the search options as default.
- **Show long description field:** The Image information panel contains two description fields: Description and Long description. Select this check box to show the Long description field.
- **Use auto page size or set it here:** Check this option if you want Object Store to automatically adjust the number of hits displayed, or set the number of hits a search will show per page. If the number of hits exceeds the page size, Object Store creates new pages.
- **Set Password:** Click the **Set Password...** button to change the Object Store password. A dialog box appears asking for the current password. Enter and confirm the new password and click the **Apply** button or **Cancel** to dismiss any changes. See also Password and Access Control.

Password and Access Control

If a password has been set for restricting access to the settings (that are saved to the Viz Pilot database), an access control dialog box will appear asking for a valid password. Restrictions apply to all settings except General settings which let users customize the appearance of the Object Store client.

Note that access control is by default disabled; however, it can be enabled by setting a password. The same dialog is used to change the password, but not to disable it, which is done from Director's **Preferences** window.

Enabling Access Control

1. Start the Object Store client.
2. Click the **Settings** button on the toolbar to open the **Settings** window.
3. Click the **Set Password...** button under the **General** tab to open the **Change password** dialog.
4. Enter the new password, confirm it, and click **Apply**.

Disabling Access Control

1. Go to the Pilot Data Server Web Interface.
2. Click the **Settings** tab.
3. Click the **VCP Parameters** tab.
4. Select the **VOS_PASSWORD** and delete the encrypted password in the Value column.

> ⊗ **Caution:** The VCP Parameters tab displays various database settings related to different Viz Pilot system applications. All changes are effective once applied.

13.9.3 Keywords, Additional Fields, Organizations and Countries
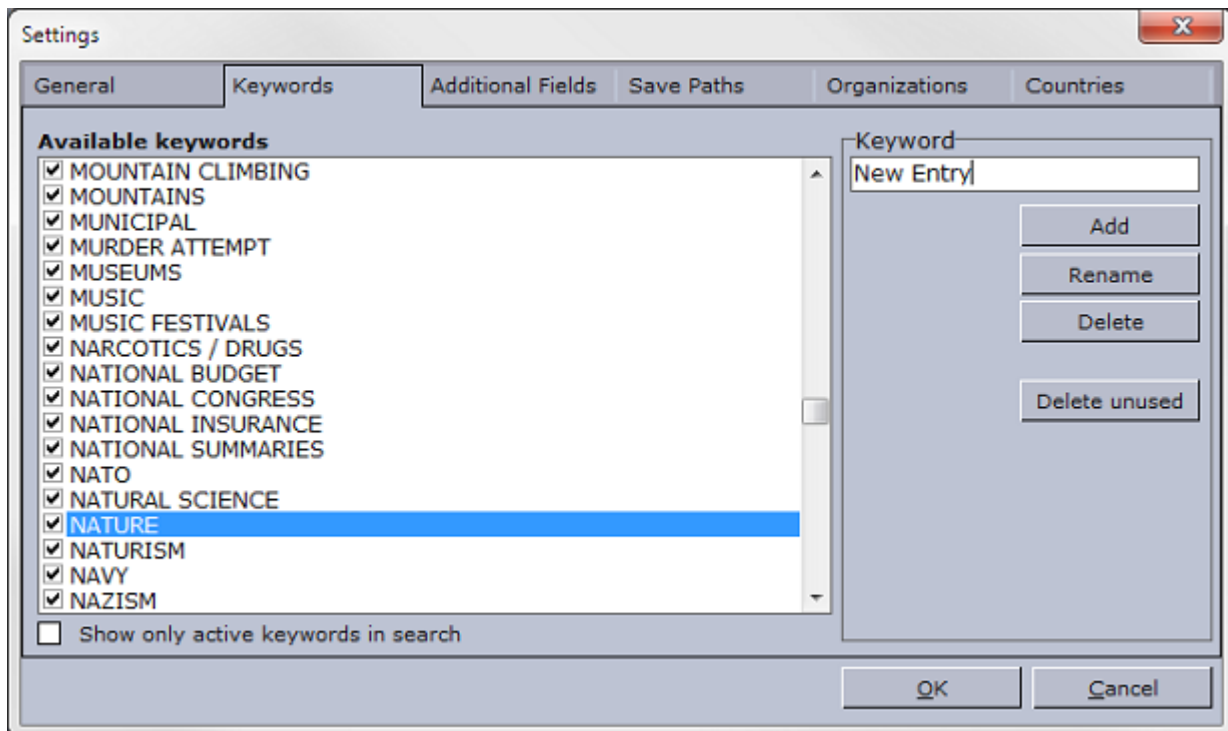
The Keywords, Additional Fields and Organizations settings all have similar functions. To access these tabs, click the **Settings** button on the Object Store Toolbar.

Actions

- **Add:** Adds a new entry. If an entry already exists the Add button is automatically disabled.
- **Rename:** Renames the selected entry. If an entry already exists, the Rename button is automatically disabled.
- **Delete:** Deletes the selected entry.

> ⚠ **Note:** Note that deleting an entry will remove it from the list, as well as all still images using it.

- **Delete unused:** Deletes unused entries. Does not apply to countries.

> ⚠ **Note:** It's not possible to delete unused entries in the Countries category.

Keywords

Keywords are used to categorize images. Although Object Store provides a default set of categories, new categories can be added and old categories can be deleted and renamed. Images or persons registered with keywords that are deleted will lose the keyword. Images or persons with keywords that are renamed will receive the renamed keyword.

☑ WORLD CHAMPIONSHIP ☐ WORLD CHAMPIONSHIP

Available keywords for new image registrations can be selected. Disabled keywords can still be used in searches. To disallow disabled keywords from searches, select the "*Show only active keywords in search*" option.

> ⚠ **Note:** New keywords are all saved in upper case.

Additional Fields

In addition to the default set of information fields provided, users often need additional customizable fields. To add new fields, click the **Settings** button on the toolbar and select the **Additional fields** tab.

☑ Weight ☐ Weight

Available additional fields must be checked for new image registrations. Disabled additional fields are not available for new registrations, but are still available as search options.

When importing images from other Object Store databases, users may find that field names differ from one Object Store installation to another. To overcome this issue, Object Store adds new field names according to the XML file used when importing images. By default, all new field names are enabled in the Image information panel.

Organizations

Organizations are used as part of Person information to show what organization a person belongs to. Add new organizations by using this setting.

☑ UNITED NATIONS ☐ UNITED NATIONS

Checked organizations are available when registering new persons. Unchecked organizations are not available for new registrations, but are still visible as search options.

The registered organization will be lost when the Person information is edited and saved if the organization is not available for new registrations. The organization information is visible to the user until the Person information is edited and saved.

> ⚠ **Note:** New organizations are saved in upper case.

Countries

Countries are used as part of Image information to set the country in which the image was taken. Use this setting to add, rename and delete countries.

Northern Mariana Islands
Norway
Oman

All countries are available at all times. Countries that should not be shown as part of the list must be deleted.

13.9.4 Save Paths
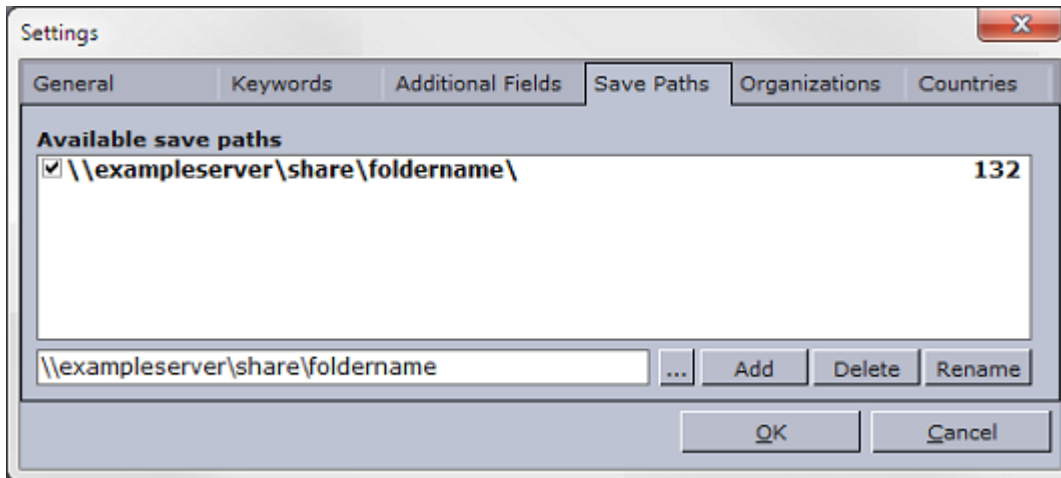
Save paths are paths to directories where images are stored, preferably on a shared file server. Object Store can add, rename and delete save paths. The number of images stored in each save path is indicated in the right-hand column.

When images are added to Object Store, a thumbnail image and data related to the image is saved in the database (see Database Configuration). The original full-size image is saved according to the Save Path setting. All images that are registered to the database will store the save path as metadata.

To access the **Save Paths** tab, click the **Settings** button on the Object Store Toolbar.

Working with Save Paths

> ❌ **IMPORTANT!** When using a remote server for storing data, the Universal Naming Convention (UNC) name must be used. Therefore, the save path set in the Object Store client should use the UNC path rather than the mapped-drive letter. For example: `\server\share\foldername` .

- **Add:** Adds a save path.
  To add a save path, browse for the folder you want the still images to be stored in and press **Add**.
- **Delete:** Deletes a save path.

> ⚠ **Note:** Save Paths with stored images cannot be deleted.

- **Rename:** Renames a save path. This is useful if a file share is moved to a new location.

> ⚠ **Note:** When moving from old VOS to RestVOS, you must rename the path in order to swap from a mapped drive to a UNC path.

To rename a save path, browse for the folder of the new location where the still images are stored and press **Rename**.

Active and Inactive Save Paths

Multiple save paths can be added to keep track of the available paths, but only one path can be active at a time. **To activate** a new save path, click the checkbox next to it and confirm the change of the active path.

> ⚠️ **Note:** Although it's possible to use images from multiple mapped sources for playout, it's recommended to store all images on a single shared file server.

When a save path is not selected as the active path, the images stored on it are still searchable in Object Store; however, it is not possible to add or edit images to/from a deactivated save path.

Save Paths in VOS and RestVOS

When using conventional VOS (not through the Pilot Data Server) the save path needs to be accessible from both the Viz Pilot system's clients and the Viz Engines.

When using Rest VOS (see the Pilot Data Server Administrator Guide), the save path only needs to be accessible from the Pilot Data Server. The Pilot Data Server then sends the image data to the client who requested it.

13.10   Object Store Information Panels

This section explains how to add, view and edit information about still images (stills) or persons in Object Store.

Image Information	Person Information	Export/Import
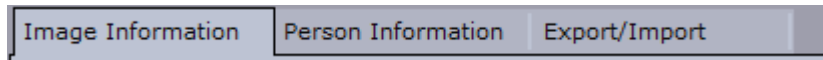
By default, the information panel consists of the tabs Image Information and Person Information. These can be enabled or disabled through the Object Store Settings window.

The Object Store Export and Import tab appears in the information panel when clicking the **export** or **import** buttons on the Object Store Toolbar.

This section contains the following topics:

- Image Information
- Image Preview
- Working with Images
- Person Information
- Working with Person Information

13.10.1   Image Information

The Image Information panel displays the currently selected image and its metadata. Some fields are auto-generated by the system, others are auto-filled based on metadata appended to the image itself, while some data is added manually. Manually added data can often be keywords added to categorize the image and to filter the search. Required fields are highlighted with a pale yellow color.

Object Store reads image metadata attributes defined by the International Press Telecommunications Council (IPTC) and automatically adds them to the appropriate information fields.

A default set of fields is shown below. Add new fields to the **Image information** panel in the settings for Keywords, Additional Fields, Organizations and Countries.

Auto-generated fields:

- **Image ID:** Sets the ID for the file in the database.
- **Directory:** Sets the file location based on the Save Paths setting.
- **Filename:** Shows the actual file name.
- **Key-Filename:** Sets a separate filename for matching stills that are keyed.
- **File date:** When the image was last modified.
- **Image size:** In pixels.
- **Description:** Sets the filename by default if no other information is embedded with the file.
- **Date registered:** Sets the date for when the still was first registered.
- **Date modified:** Sets the date for when the still was last modified.
- **Keywords:** Sets the keyword for the still image. One keyword is required. If no appropriate keywords are available, new ones can be added in the Object Store Settings window.
- **Person:** Sets the link to Person Information. Only one Person can be linked.
- **Source:** Sets the name of the copyright holder or image origin.
- **Country:** When selected, gives the still a geographical location by country.
- **City:** When entered, gives the still a geographical location by city.
- **Registered by:** Sets the logged-in user as the registrant when information is saved.

- **Approved:** When set, indicates that the still is ready for use.
- **Additional fields:** Displays additional fields added in the Object Store Settings window.
- **Image:** Displays the uploaded image. Double-clicking the image will open the Image Preview.

13.10.2   Image Preview

Once an image has been selected and is displayed in the Image Information panel, it can be previewed by clicking **Preview** (**CTRL + P**) in the Object Store Toolbar, or double-clicking the thumbnail in the **Image Information** panel.

- **Clipboard:** Puts the image on the clipboard allowing the image to be pasted into other applications.
- **Key & RGB:** Shows the key and RGB version of the image. If the image has no key, an all white image (full key) is shown. Click **Key** to replace it with the **RGB** button.

- **Blended:** Shows the key image blended with a white background.
- **Crop:** Opens the **Crop** image window.
- **Save:** Saves a copy of the image on the computer. The following formats are available: Bitmap, JPEG, RGB, Targa, TIFF and PNG. By default the format is set to TIFF.
- **Close:** Closes the Preview dialog box.

> ⚠ **Note:** The RGB, Key & Blended views are also available as part of the Image Information.

Previewing an Image

1. Search for an image.
2. Double-click the thumbnail to open the **Information** panel.
3. To preview the image click **Preview** (**CTRL + P**) in the Object Store Toolbar, or double-click the thumbnail in the **Image Information** panel.

> ⚠ **Note:** Stills are shown in full size if full size does not exceed the screen resolution.

13.10.3   Working with Images

Adding a New Image

1. Click the **New Image** button on the Object Store Toolbar, or use the keyboard shortcut **CTRL+N**.
2. Browse for and select an image from a local drive or a network drive, and click **Open**.
3. Fill in image information.
4. When done, click **Save** (**CTRL + S**).

Editing Image Information

1. Search for the image using Still Search.
2. Double-click to open the selected image in the Image Information panel.
3. Click **Edit** (**CTRL + E**) on the Object Store Toolbar to enable editing.
4. When done, click **Save** (**CTRL + S**).

Linking a Person to an Image

1. Open an image and make it editable.
2. Click the ellipse button ⬚ to search for a new person using Person Search.
3. Double-click the list item representing the new person to replace the old Person information link.
4. When done, click **Save** (**CTRL + S**).

> ⚠️ **Note:** Only one person can be linked to an image at a time. This is a limitation that relates to how portrait images are linked to person information.

13.10.4   Person Information

Required fields:

- **First name**
- **Last name**
- **TV name:** The name the TV station uses.

Optional fields:

- **Middle name**
- **Title:** Can be used to add a title to the person (for example Lady, Sir, Dr., Mr., Mrs. etc.).
- **Comment:** Can be used as an open text field describing the person.
- **Organization:** Can be used to link the person to an organization.
- **Place:** Can be used to place the person geographically.
- **Images Linked:** Can be used to link images to a person.
- **Bin (icon):** Removes images that are linked to the person.
- **Save:** Saves new or updated person information to the database.

13.10.5   Working with Person Information

Adding Person Information

1. Click the **New Person** button on the Object Store Toolbar.
2. Fill in the person information.
3. When done, click **Save** (**CTRL + S**).

Editing Person Information

1. Search for a person using Person Search.
2. Double-click to open the selected person in the Person Information panel.
3. Click the **Edit** button (**CTRL + E**) on the Object Store Toolbar to enable editing.
4. When done, click **Save** (**CTRL + S**).

Deleting Person Information
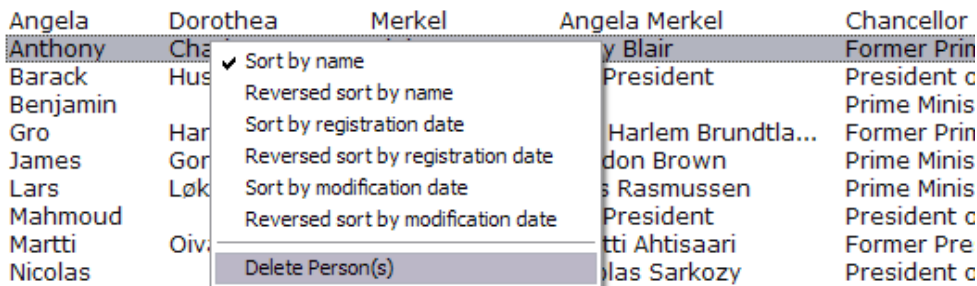
1. Search for a person using Person Search.
2. Right-click the person list entry, and select **Delete Person(s)**.
3. Click **Yes** to confirm the delete operation.

> ❌ **Caution:** Deleted persons from the database cannot be restored. Images linked to the person will also be permanently deleted.

Linking Images to a Person
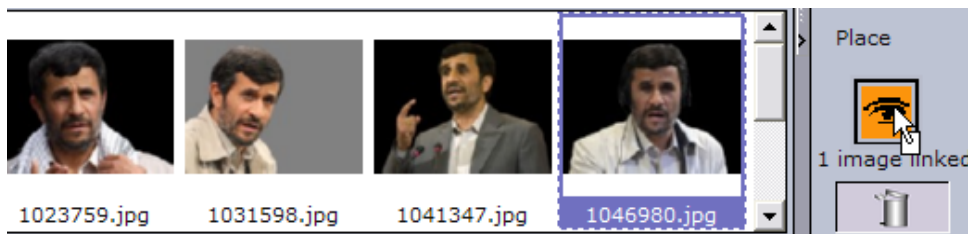
1. Search for a person using Person Search.
2. Double-click to open the selected person in the Person Information panel.
3. Click **Edit** (**CTRL + E**) on the Object Store Toolbar to enable editing.
4. Search for images using Still Search.
5. Drag and drop images onto the eye icon visible in the Person Information panel.
6. When done, click **Save** (**CTRL + S**).

Unlinking Images to a Person
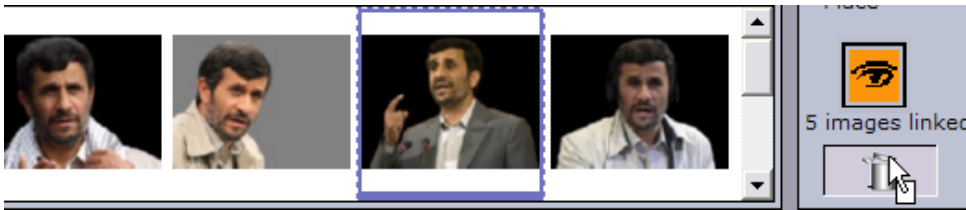
1. Search for a person using Person Search.
2. Double-click to open the selected person in the Person Information panel.
3. Click **Edit** (**CTRL + E**) on the Object Store Toolbar to enable editing.
4. Drag and drop images that are to be unlinked into the bin in the Person Information panel.
5. When done, click **Save** (**CTRL + S**).

Adding an Image to a Person

1. Search for a person using Person Search.
2. Double-click to open the selected person in the Person Information panel.
3. Double-click the **eye icon** to open the file browser.
4. Browse for and select an image from a local drive or a network drive, and click **Open**.
5. Fill in image information, and click **Save** (**CTRL + S**) to return to the Person Information panel.
6. When done, click **Save** (**CTRL + S**).

See Also

- Image Information
- Organizations
- Object Store Export and Import
- Object Store Crop Tool

14 Thumbnail Generator

Thumbnail Generator is a Windows service that automatically generates thumbnails of data elements as they are created or changed. The thumbnails are stored in the Viz Pilot database.

The thumbnails are used in the data element lists in Viz Pilot News and Director, as well as the Legacy Playlist, but are not used in the new Playlist Window. They are not used in the Pilot Data Server either, since the Pilot Data Server uses thumbnail links that point to the Preview Server in its data element descriptions.

For information on installing Thumbnail Generator, see Pilot Data Server Installation.

14.1 Configuration

Only one instance of Thumbnail Generator should be running at any given time. Having two instances running will not lead to errors, but will create an extra load on the Preview Server and its Viz Engine, since all thumbnails will be generated twice.

The Preview Server must be configured in the Pilot Data Server under **Settings > VCP Parameters >** `preview_server_uri` . See the *Viz Pilot Release Notes* for the recommended versions of Preview Server.

Thumbnail Generator uses the database specified during Pilot Data Server installation. To change the database, run the Pilot Data Server Installation again and select **Change**, or click **Change** from the list of installed programs in Windows.

14.2 Operation

Thumbnail Generator listens to database change notifications in order to update thumbnails automatically. It maintains a queue of data elements to be processed, which will be ordered so that the data elements most recently updated will be processed first. If there are many simultaneous changes, it may take some time before the thumbnail of a specific element is generated by Thumbnail Generator. To avoid overloading the Preview Server or the Viz Engine with thumbnail requests, Thumbnail Generator will also wait a number of seconds between consecutive requests.

Thumbnail Generator starts automatically when Windows starts. Thumbnail Generator handles database connection issues and doesn't require any kind of maintenance.

14.3 Logging

Thumbnail Generator places its log file at:

```
c:\ProgramData\Vizrt\Logs\Viz Pilot Thumbnail Generator\Thumbnail Generator.log
```

When Thumbnail Generator fails to generate a thumbnail for a data element it will retry with increasing intervals. If it fails a certain number of times, the item will not be retried any more, leaving the data element without a thumbnail.

15   Database Administration

The database installation for Viz Pilot is a typical Oracle database server installation.

Vizrt's standard installation of a Viz Pilot database uses the service name **VIZRTDB**. The SQL schema and table creation scripts add the user PILOT with password PILOT which make up the schema name and the default credentials that all Viz Pilot system applications are configured to use. This can be changed either during or after installation.

15.1   User Interface

Starting the client opens the Connect to DB window - the initial screen of Pilot DBA. This window contains: the Database Login, different Management Options and Advanced Configuration of user levels.

This section covers the following topics:

- Connect to DB
- Database Login
- Advanced Configuration
- Schema Status
- Connection Status
- Database Error Logs

15.1.1   Connect to DB

The following options are available from the initial window:
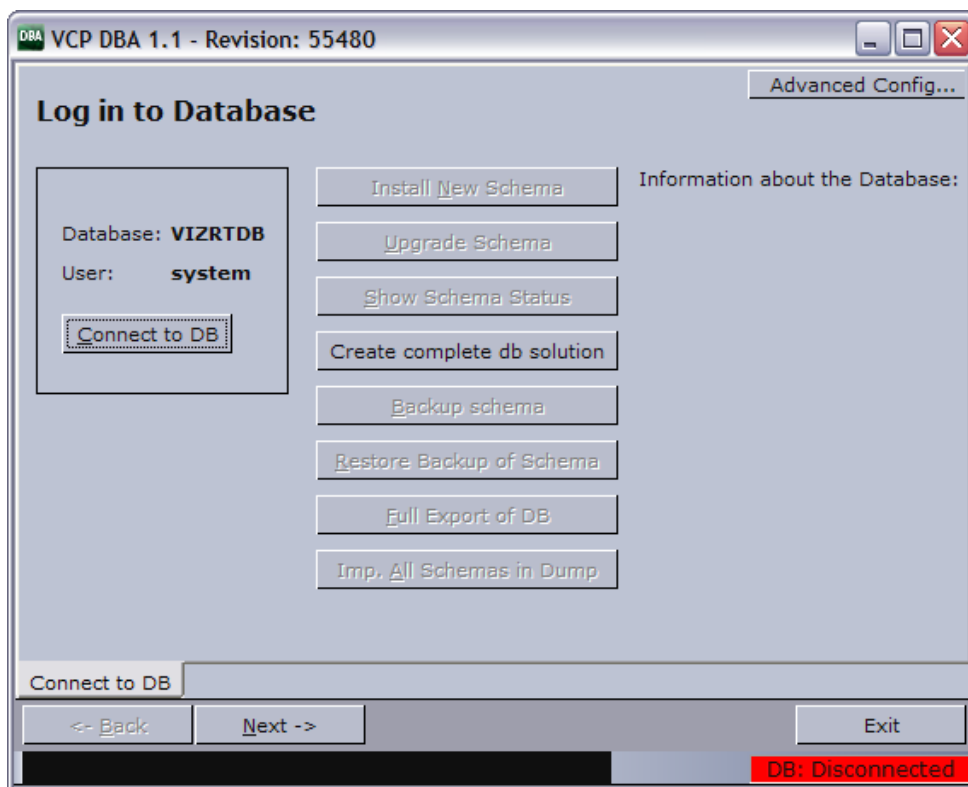
- **Connect to DB/Change DB:** Lets you log on to the database.
- **Advanced Config ...:** Lets you select a default user level or create a custom user level.
- **Install New Schema:** Lets you install a schema on the specified database. See Install Schema.
- **Upgrade Current Schema:** Lets you upgrade the current version of the schema to a later version. See Upgrade Schema.

> ⚠️ **Note:** The application first scans the installed schema for critical errors and blocks the option to upgrade if any critical errors are found.

- **Show Schema Status:** Lets you see the status of the currently installed schema. See Show Schema Status.
- **Create complete db solution:** Lets you user create a complete database solution. See Create Complete DB Solution.
- **Backup Schema:** Lets you export a database dump of the schema. See Backup Schema.
- **Restore Backup of Schema:** Lets you import a database dump of a schema. See Restore Backup of Schema.
- **Full Export of DB:** Lets you perform an export of the entire database. See Full Export of Database.
- **Imp. All Schemas in Dump:** Lets you import all schemas in dump. See Import All Schemas in Dump.

15.1.2 Database Login

In order to successfully connect to a database, a local net service name must be configured for the Oracle database client using Oracle's Net Configuration Assistant.
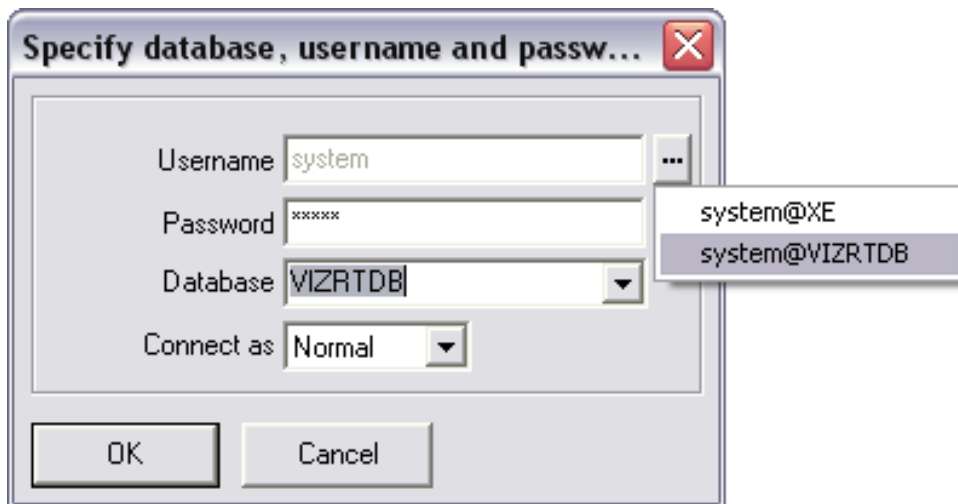
- **Username:** Displays the pre-configured system user.
- **Password:** Sets the password for the system user.
- **Database:** Sets the database service name or connection string.
- **Connect as:** Sets the connection privileges. Available options are Normal, SYSDBA or SYSOPER.

15.1.3   Advanced Configuration
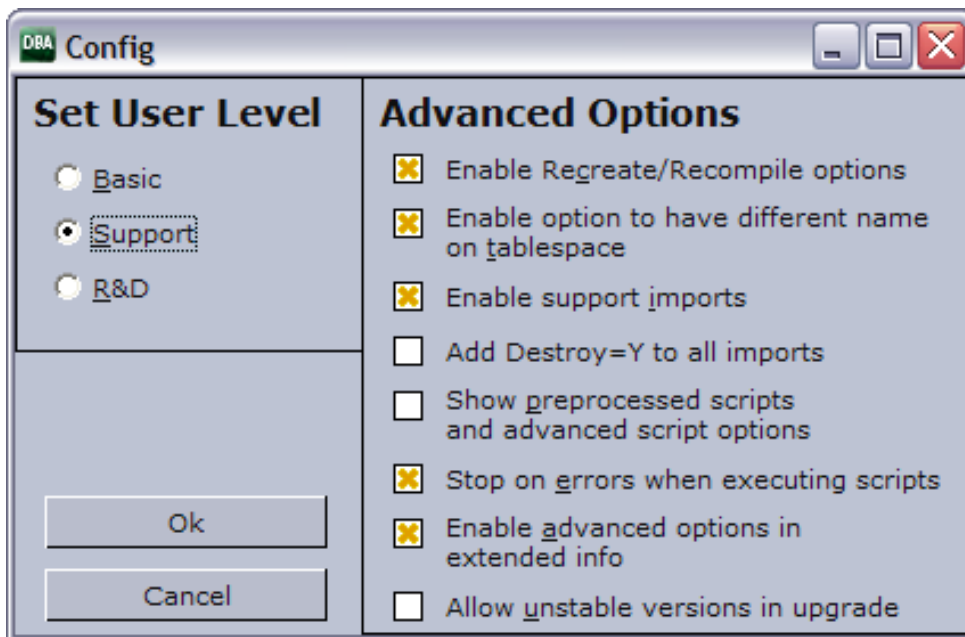
- **Set User Level:** Shows the available user modes.
  - **Basic:** This level is for users who are doing regular maintenance and fresh installations or upgrades of the database schema.
  - **Support:** This level is mainly used by Vizrt's support personnel and skilled administrators who are doing special customizations to an existing installation or fixing common errors found in schemas.
  - **R&D:** This level is for use by Vizrt's R&D department when doing development or searching for bugs that are not resolved by customer support.
- **Advanced Options:** Shows the available options.
  - **Enable Recreate/Recompile options:** This enables the Recreate/Recompile options in the Schema Status window and the button to recreate AQ tables in Extended Info-AQ & Purge Job.
  - **Enable option to have different name on tablespace:** Enables the user to select a different tablespace name. Default is the same as Schema name.
  - **Enable support imports:** Enables a "support only" version in the combo box for selecting a version for import. This is for importing a single schema without first installing a new schema, so that Vizrt can debug the schema. Also enables "Imp.All Schemas in Dump" option.
  - **Add Destroy=Y to all imports:** Overwrites tablespaces if already found.
  - **Show preprocessed scripts and advanced script options:** Shows an extra tab before executing the chosen script where the user can look through the pre-processed script or reload it.
 - **Stop on errors when executing scripts:** Stops the execution of scripts when errors occur. For versions 5.1.7 and later this option should be enabled. For older versions than 5.1.7 this option should be disabled.
  - **Enable advanced options in extended info:** Enables the buttons to activate/deactivate the AQ triggers and purge job, and the button to delete all entries in the errlog table older than 3 months.
  - **Allow unstable versions in upgrade:** Allows the schema to be upgraded to/installed as an unstable version; intended for use by Vizrt staff only.

15.1.4   Schema Status
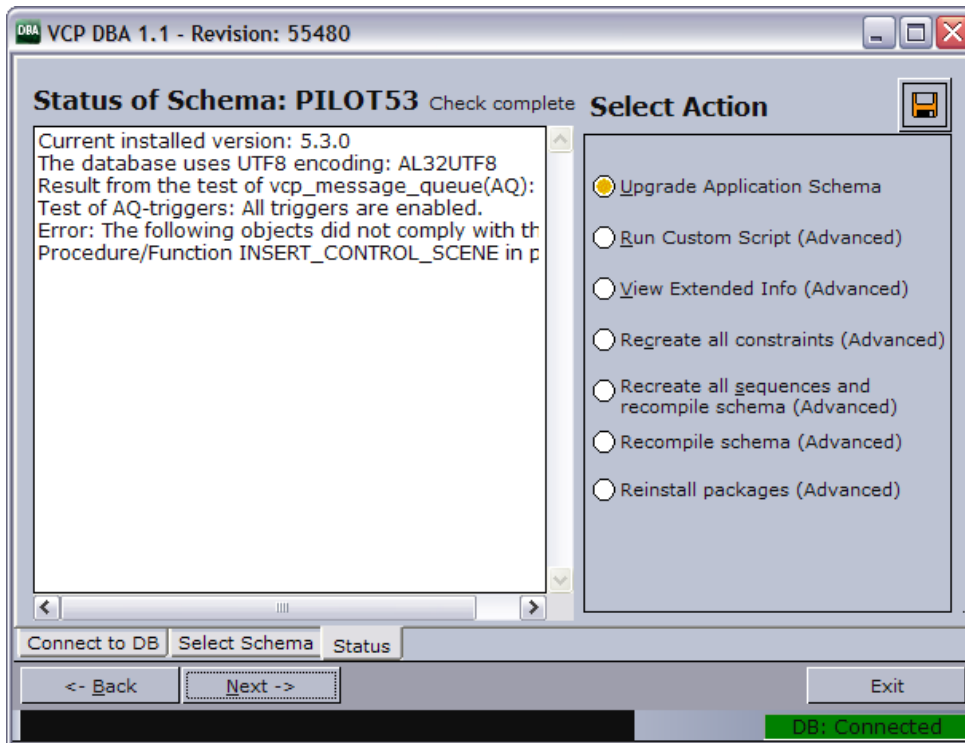
- **Save icon:** Creates a packaged file (zip) with debug information. The file can be sent to support for external help with the database.

> ⚠️ **Note:** If dump files are not created as expected, check that Pilot DBA's installation path does not contain spaces.

- **Upgrade Application Schema:** Lets the user upgrade to a new schema.
- **Run Custom Script (Advanced):** Lets the user run a custom script specific to the database.
- **View Extended Info (Advanced):** Lets the user display extended information about the schema.
- **Recreate all constraints (Advanced):** Lets the user execute a script that recreates all constraints in the schema. Typically needed after a "support only" import, or if some constraints are missing. Missing constraints are reported in the status window.
- **Recreate all sequences and recompile schema (Advanced):** Lets the user recreate all sequences in the schema and then recompiles the entire schema. This is usually done by normal imports, but could be needed if something went wrong during the import or if "support only" import was used. This will also fix problems that might appear if a manual import was done.
- **Recompile schema (Advanced):** Enables the user to recompile all packages, triggers, functions and procedures.

15.1.5 Connection Status

`DB: Connected` `DB: Disconnected`

The connection indicator field at the bottom right shows the connection status between Pilot DBA and the database: red when disconnected, and green when connected. If the connection is lost, try clicking on this field to reconnect to the database.

15.1.6 Database Error Logs

`Error!`

When errors occur during operation, an indicator is shown in the lower right of Pilot DBA's user interface. Clicking this warning indicator will open a log window containing the following three logs:

- **VCPDBA.log:** Shows the main application log.
- **VCPDBA_SQL.log:** Shows the SQL output from the last script that was executed.
- **vcp_ora.log:** Shows the log for the internal Oracle module.
  All log files are stored in the same directory as the Pilot DBA application.

> ⚠️ **Note:** All `vcp_ora.log` files are added to the `vcp_ora.log.old` file.

Database Log Files

The alert logs for the Oracle Viz Pilot database are generally found in the following locations, although this may differ slightly on your system:

- `C:\oracle\diag\rdbms\<dbname>\<SID>\alert\log.xml`
- `C:\oracle\diag\rdbms\<dbname>\<SID>\trace\alert_<dbname>.txt`

See Also

- Database Service Names and SID
- Database TNS Alias
- Newsroom workflow without database connection

15.2   Database Limitations

15.2.1   Current Limitations

- Pilot DBA's install path cannot contain spaces.
- Pilot DBA can only login with Oracle's system user.

15.2.2   Upgrade Considerations

As a general rule, the schema version should be upgraded to the same version as Director. It is important to remember the order in which to do the upgrade, and to do the schema upgrade work at a convenient time when the system is not in use.

Please refer to the Viz Pilot release notes for recommended schema version for specific versions of the Viz Pilot system.

As the order is crucial for the functioning of the templates in the Viz Pilot system, the following order of upgrading is recommended. If instructions are not followed and a newer version of Template Wizard is used to save a template, the template will most likely not work inside Viz Pilot News or Director.

1. Upgrade Viz Pilot system's schema in the Viz Pilot database.
2. Upgrade all clients except the Template Wizard clients.
3. Upgrade the Template Wizard clients.
   The order of 1 and 2 can be switched, but the last component to be upgraded should always be the Template Wizard. This is to avoid saving down templates with new functionality that may come with the version, which is not supported by older versions of Viz Pilot News or Director.

Test the versions of Viz Pilot News, Template Wizard and Director you want to upgrade on one non-live client of each at minimum. The test should at minimum include opening all templates in each client, saving down one data element of each template in Viz Pilot News, and playing all those data elements out from the updated Director.

Contact your local support team with information about the current state of the Viz Pilot database; for example, which version the current Viz Pilot schema is on and what version you want to upgrade to so that support is aware that the upgrade is happening and can prepare second line and personal follow-up, if required.

When you start the upgrade process, any data elements or new templates saved after this time may be lost if the upgrade is not successful. The chances of this are small, but it may happen and it should be incorporated into the plans.

If you have a main and standby setup, check that they are synchronized. If this is not the case, Vizrt support must be informed and any schema upgrade cannot go ahead before this is rectified. If it cannot be solved, a full recreation of the standby server may be needed before continuing. Also, a set time is required for when you can break the synchronization between the two databases.

15.2.3   Upgrade Limitations

When upgrading from a Viz Pilot 4.x or older version it is advisable to contact a local Vizrt representative or support person so they can advise you on how to proceed. Some features found in Viz Pilot 4.x are not available in later versions.

When upgrading from a Viz Pilot 5.x version it is advisable to upgrade the database schema to the latest schema version that is compatible with the software version. When upgrading the database, please use the Database Administration tool. For more information on supported schema versions, see the release notes.

15.3   Database Connection Parameters

Pilot DBA (Pilot Database Administrator tool) is used by database administrators to maintain and upgrade the Viz Pilot database system. It also enables the database administrator to check the current status of the database.

Parameter	Value
Oracle system identifier (SID/Service name)	VIZRTDB
All system users (default)	system/oraclesys/oracle
All users (default)	PILOT/PILOT
Database schema (default)	PILOT

> ❌ **Important!** Vizrt does not support installations on express edition (XE) databases.

15.3.1   Getting Started - Pilot DBA

1. Browse to the location where you extracted the Pilot DBA files.
2. Double-click the Pilot DBA executable file to start the application.

> ⚠ **Note:** Make sure Viz Engine is configured and running.

See Also

- For information on installing an Oracle database server, see the documentation supplied by Oracle.

15.4   Management Options

This section describes the different database management options available in the Pilot DBA User Interface:

- Install Schema
- Upgrade Schema
- Show Schema Status
- Creating a Complete DB Solution
- Backup Schema
- Restore Backup of Schema
- Full Export of Database
- Import All Schemas in Dump
- Extended Information
- Database Test

15.4.1   Install Schema

Install a new schema, set a schema name, username and password, and a tablespace name. If the user level is set to R&D mode (see Advanced Configuration), it's possible to install an unstable version of the database schema. This is typically the latest version in development, and is only recommended for testing.

Installing a New Schema

1. Click the **Install New Schema** button.
2. In the New Schema window, select the Viz Pilot version to be installed.
3. *Optional*: Enter the schema name and password
4. *Optional*: Enter the tablespace name. Setting a different tablespace name requires this option to be selected (see Advanced Configuration).
5. Click **Next** twice to start the installation.
6. Click **Next** when finished.

> ⚠️ **Note:** If errors occur, see Database Error Logs.

15.4.2   Upgrade Schema

Upgrade to to a newer version of the schema. If the user level is set to R&D mode (see Advanced Configuration) it's possible to upgrade to an unstable version of the database schema. This is typically the latest version in development and is only recommended for testing.

Upgrading to a New Schema

1. Click the **Upgrade Schema** button.
2. In the Select Schema window, select the schema that should be upgraded, and click **Next**.
   - When Pilot DBA has checked the status of the current schema, the Upgrade Schema window lists the available upgrade options.
3. Select the schema version, and click **Next** to see the script that will be executed.
4. When ready to upgrade, click **Next** to run the scripts.

5. Click **Next** when finished.

> ⚠️ **Note:** If errors occur, see Database Error Logs.

15.4.3 Show Schema Status

The list shows the status of the following:

- The version of the currently installed schema.
- Text encoding of the database.
- Status of AQ.
- Whether the schema complies with a reference of the currently installed version.
  If more than one schema is installed on the same database, a selection dialog is displayed before the status information window is shown.

The available options depend on the configured user level, see Advanced Configuration. Note that most features will be blocked if the database text encoding is not UTF-8.

> ❌ **Caution:** With the exception of **View Extended Info**, the **Advanced** options may stop the schema from working if an error occurs while running the scripts.

Viewing Schema Status

1. Click the **Show Schema Status** button.
2. Select the schema and enter the schema password.
3. Click **Next**.

15.4.4 Creating a Complete DB Solution

Creating a complete database solution using Pilot DBA ensures that the database is set up and configured according to our recommendations. It also installs the latest version of the PILOT schema.

- **The Oracle Standalone Server** is a setup with a single server and no form of automated backup.
- **The Oracle Standby Server** is a setup with two standard Oracle servers where one is working as the failover server in case the main server goes down. The standby server's failover mechanism is manual, and the standby server is inactive for users when in standby mode. The standby server setup requires that the main server is started before the backup server.

> ❌ **IMPORTANT!** There is no instant synchronization between the servers.

> ⚠️ **Note:** The standby setup is meant to be used with new installations of standby servers. It is created to work with C: drive as the program disk and D: drive as the data disk.

Creating a Complete Stand-alone Server

1. Click Advanced Configuration and select the support or R&D user level.

2. Click the **Create complete db solution** button.
3. Select the **Stand Alone Server (Windows)** option.
4. Select the drives for the software and the data files.
5. Enter the **hostname** if the hostname was not detected.
6. Click **Next**.
   - Clicking **Next** will launch the Oracle Universal Installer, and a separate command shell. The Oracle Universal Installer will install the database.
   - After the installation the Oracle Universal Installer will start to execute the configuration assistants (for example Oracle Net Configuration Assistant and so on).
7. In the separate command shell window, click **any key** to continue after receiving the message " `The installation of Oracle Database 11g was successful.` " from the Oracle Universal Installer.
8. Pilot DBA will now log itself on to the database and install the **PILOT schema**.
9. Check that the host information in **listener.ora** and **tnsnames.ora** use **hostnames** and not IP-addresses.

> ⓘ **Example:** `C:\oracle\product\10.2.0\db_1\NETWORK\ADMIN\`

Create a Complete Standby Server

- It is not recommended to perform this setup option without consulting your local Vizrt representative or support person.

15.4.5 Backup Schema

The database backup feature exports a database dump of the schema. If the backup does not produce an output, it is important to check that the Pilot DBA install folder does not contain spaces, and that the Oracle Administrator client is used.

15.4.6 Restore Backup of Schema

Restore a backup imports a database dump of a schema. It is possible to choose to import to a new schema if the schema name imported to is not the same as the schema name that was exported; however, this also requires that the version of the schema that was dumped is known. The application will perform a complete install of that schema version, truncate the tables and import the dump.

Setting the user level to Support will enable the option to select the Viz Pilot version Support only. This version allows a raw import of a database schema which in turn can be used for debugging. A schema imported this way will not have AQ, triggers or constraints enabled and will therefore not be a schema one can use with Viz Pilot system applications.

15.4.7 Full Export of Database

Full Export of Database does an export of the entire database, including all schemas in the database. This is mainly used for Customer Support Requests.

The database dump file is saved in the Pilot DBA installation folder, for example `exp_complete.dmp` .

15.4.8   Import All Schemas in Dump

Import All Schemas in Dump analyses a dump file to find all tablespaces and users before creating them and importing data from the dump file. Oracle default users and tablespaces are excluded from the imports.

This option is only available to Support and R&D users.

15.4.9   Extended Information
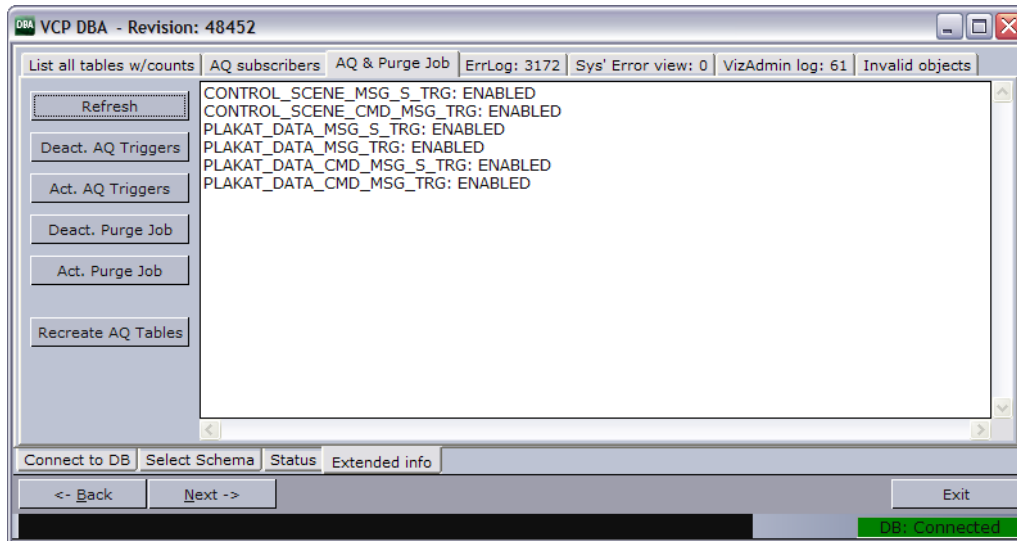
Extended information can be viewed if the option **Enable advanced options in extended info** is enabled.

> ⊗ **Caution:** The different actions in these views could potentially stop the schema from working if an error occurs while running the scripts.

Information Tabs

All information tabs have the option to refresh the database information result by clicking the **Refresh** button.

The following information and options are available:

- **List all tables with counts:** Lists all tables in the schema and the number of rows in each.
- **AQ Subscribers:** Shows a list of the message queue (AQ) Subscribers currently registered.
- **AQ & Purge Job:** Lists the status of and allows the user to activate/deactivate the AQ Triggers and the Purge Job. This view also has an option to recreate the AQ tables, which is done by deleting and recreating the tables.
    - **Deact. and Act. AQ Triggers:** Disables and enables the AQ system if this is working properly.
    - **Deact. and Act. Purge Job:** Disables and enables a scheduled job on the database that purges old messages from the AQ system to avoid these tables clogging the database with obsolete data. (see Purge Expired Messages).
    - **Recreate AQ Tables:** Deletes all AQ tables and all data in them, and then creates new tables and registers these with Oracle's AQ system. This option will **delete** all messages not received in the AQ

tables; the AQ system will not work afterwards if any errors occur. This option should therefore not be used by users that do not know how to read error logs and who are not able to fix AQ issues manually.

- **Errlog: (# items in the errlog):** Lists the last 100 entries in the errlog and lets users optionally delete all entries older than 3 months.
- **Sys' Error view:** Lists all errors with the current schema as owner in the sys' all_errors view.
- **VizAdmin log:** Shows the last 100 entries of the VizAdmin log table.
- **Invalid objects:** Shows all invalid objects in the database.

Purge Expired Messages

A MOS message is sent from the database to Viz Gateway when a journalist creates a new element of a template. All MOS messages are stored in the database which in turn takes up space on the database server. It is therefore a good idea to purge expired messages from the Viz Pilot system's message queue. Expired messages must be explicitly removed since they are transferred to the exception queue. This can be done by running a job at regular intervals (every day). This is done every day by default.

See Extended Information

1. Click__**Show Schema Status**.
2. Select schema, enter schema user password, and click **Next**.
3. Select **View Extended Info (Advanced)**, and click **Next**.
4. Select one of the tabs to view the extended information.

15.4.10   Database Test

Try connecting to Viz Pilot's database to test if the database is successfully installed and running.

Start a command-line tool, and then type the following:

- sqlplus userid/password@tnsname, *or*
- sqlplus userid/password@hostname/SID
  SQLPlus connection:

Check that the output is similar to the example output below:

```
C:>sqlplus pilot/pilot@viz-xw4300/xe SQL*Plus: Release 10.2.0.1.0 - Production on Wed
Jun 20 12:17:19 2007 Copyright (c) 1982, 2005, Oracle. All rights reserved. Connected
to: Oracle Database 10g Express Edition Release 10.2.0.1.0 - Production SQL>
```

The example output above shows that the database is placed on a remote machine, named *viz-xw4300*, thus the connection string *@viz-xw4300/xe* (*hostname/SID*) or the TNS name only is required.

```
C:>sqlplus pilot/pilot@VIZRTDB
```

> ⚠ **Note:** The above screenshot and output example uses the SID *XE*. This is the default SID for Oracle 10g XE, which is an Oracle test database.

See Also

- Database Administration
- Upgrade Considerations
- Upgrade Limitations

16   Appendix

This section contains:

- Storage and Backup
- Glossary
- System Overview
- Integrating Drag and Drop from External Applications
- Port Numbers
- Working Without a Database Connection
- Legacy Playlist
- Log Files

16.1   Storage And Backup

This section provides general recommendations on storage and use of control room and newsroom configuration data:

- Viz Pilot Data
- Media Sequencer Data

16.1.1   Viz Pilot Data

In addition to information stored on Media Sequencer and in Viz Pilot's database, it is important to back up the Viz Pilot system's INI files and playlists when required. INI files are used for local configurations for the application itself, third party integrations, and overrides of database settings. All Viz Pilot system playlists, both local and newsroom playlists, can be exported and imported for backup and transfer to other Viz Pilot system installations.

16.1.2   Media Sequencer Data

The file default.xml is a Media Sequencer file. The file contains a copy of information stored in the memory, control application client settings, user data for graphics templates, references to scenes on the database, stills in the still store and video clips.

For backup, it is recommended to first set up the system, then save a copy of the _default.xml_ remotely, to more easily set up a new build of a similar machine in the event of a system failure.

The default.xml file stores all persistent data. At regular intervals the Media Sequencer creates a backup file named *default.xml.1*, which is replaced in turn by the file *default.xml.2*. In a failure situation where default.xml may be corrupted, default.xml.1 is used to create a new instance of default.xml. If default.xml.1 is also corrupted, default.xml.2 will be attempted.

Microsoft Windows XP: *C:\Documents and Settings\All Users\Application Data\Vizrt\Media Sequencer.*

16.2   Glossary

The following terms are used in this guide:

- **Background Scene**: A background scene contains logic that enables an operator to toggle the states of multiple front scenes. A typical background scene also contains back plates and transition effects for each state of the front scenes. For example, a lower third may be on-air at the same time as a bug, and the lower third may be taken off-air without taking the bug off-air or reanimating it.
- **Component**: In Template Wizard, components are used to create text fields, buttons, drop-lists, database connections etc.
- **Control Plugin**: A graphics scene can contain all sorts of objects that can be controlled from a template such as text, back plates, images, colors and more. The graphics designer uses control plugins to expose objects to make them editable in a template. When importing a scene to Template Wizard, the exposed objects with control plugins appear as list items with check boxes.
- **Control Object**: Every scene with control plugins must have one instance of the control object plugin at the root level of a scene tree. Control Object reads the information from all other control plugins. When a scene is imported to Template Wizard, the information about other lower level control plugins is read by Control Object.
- **Data Element**: A data element is a template filled with data. A data element contains a reference to its template (original) just as the template contains a reference to its scene. A data element contains a set of data and references to where data such as images and video clips can be found. In most cases, it's the data element that is played out on-air, and not the template.
- **Front Scene**: A front scene is a standard scene that makes up one part of a transition logic scene.
- **Newsroom Component**: In a Newsroom Computer System (NCS), the Newsroom Component (NrC) is used to add data elements to a story. The newsroom component is an embedded application in the NCS that connects to a database of templates. The templates can be filled with text, images, video and maps. Vizrt's newsroom component is called Viz Pilot News.
- **Scene**: Scenes are built in Viz Artist. These can be either a single scene, or one part (layer) of a combination of scenes (transition logic).
- **Template**: A template is a customized interface created in Template Wizard. The template is used in Director or Viz Pilot News to create data elements that are added to a playlist for playout. A template is based on one or several (transition logic) Viz Artist scenes.
- **Transition Logic Scene**: A set of scenes built in Viz Artist. Consists of a background scene and one or several front scenes. The transition logic scene contains one scene (the controlling scene) that controls the state of or toggles a set of scenes (the layered scenes). The controlling scene toggles in and out of the layered scenes using pre-configured or customized transitions effects. This means that scenes that are already on-air do not need to be taken off air. For example, a lower third may be on-air at the same time as a bug, and may be taken off-air without reanimating or taking the bug off air.
- **Viz Artist**: The design tool where the graphics scenes and all animations are created.
- **Viz Engine**: The output engine used for playout of graphics, video, images, SDI sound and sound effects.

16.3   System Overview
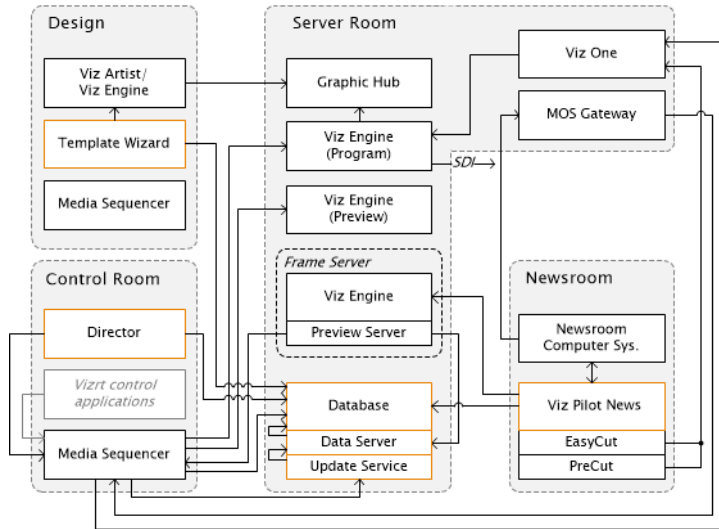
16.3.1   Workflow
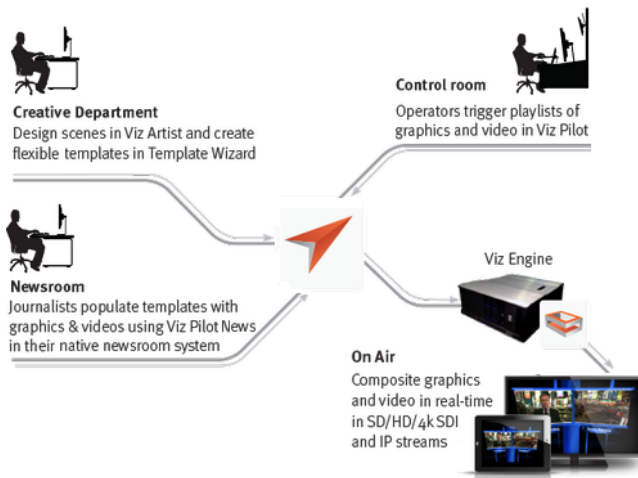
The following is a common workflow using Vizrt's MOS (Media Object Server Communications Protocol) integration:

1. A user adds stories to the newsroom system using Viz Pilot News.
2. In addition to creating stories, graphics, video clips and images can be added to the stories using Viz Pilot.
   - New data elements created by Viz Pilot are added to the Viz Pilot database.
   - The newsroom system client adds the data element information to the story.
3. Whenever a newsroom playlist is monitored by a control application, the newsroom system is requested to send information about the playlist to Viz Gateway which sends the playlist to the Media Sequencer.
4. Based on the information received from the newsroom system, the Media Sequencer is able to retrieve the data elements from the Viz Pilot database, which were created by Viz Pilot, and present them as a playlist to the control client.

- When a playlist is made available to the control client it can monitor the playlist and receive all changes to it from the newsroom system and the Viz Pilot database.
5. Once the playlist is monitored, the control client operator can take the graphics on air.

> ⚠️ **Note:** Media Sequencer stores all playlists/shows, making them available to all control applications.

16.3.2 Client-Server Model

Viz Pilot is built on a client-server software model where Director connects to the Viz Pilot Database (Oracle database or Graphic Hub) for templates and content, and to the Media Sequencer for playout.

- The Viz Pilot system can use either an Oracle database or a Graphic Hub as the back-end server.
- The Viz Pilot system requires a Media Sequencer for communication with Viz Engine and other systems. On the server side, the database serves all clients that store and receive content data for the control, delivery and playout of videos, maps, audio, graphics and so on, on the Viz Engine.

Viz Pilot News has its own Timeline Editor for adding graphics to video clips fetched from Viz One. Viz One integration enables playout of fullscreen and/or embedded video clips in graphics on the same renderer.

The overview diagram above also shows a newsroom setup using the MOS protocol (see MOS Gateway). Other protocols are also supported (Video Disk Control Protocol, for example).

16.4   Integrating Drag And Drop From External Applications

External applications can integrate with the Viz Pilot system to provide resources such as images and videos using drag-and-drop. For example, using a MAM system to locate media and using drag-and-drop to ingest the media to a template in a Viz Pilot workflow. The required data formats and drag-and-drop parameters are described below.

> ⚠ **Note:** Drag-and-drop integration in Viz Pilot is not designed to accept common image and video files that are handled by typical applications. The purpose of the integration is to let developers support drag-and-drop by API integration. Consequently, this section is fairly technical and is intended for developers who are planning to integrate drag-and-drop.

16.4.1   Formats

The integration uses the standard Microsoft Windows functions for OLE drag-and-drop, see https://msdn.microsoft.com/en-us/library/96826a87.aspx. For general information about the Windows Clipboard, see MSDN Clipboard Formats.

The drop target of a drag-and-drop operation is always an image control (TTWImage) on a template. Not every image control will accept drops. The accepted custom clipboard formats (https://msdn.microsoft.com/en-us/library/838a3whf.aspx) and what each format accepts are listed below:

Clipboard format	Accepting ImageSources	Media Type
CF_VIDEO_ASSET_ENTRY_XML	isVideoMediaSearch	video
CF_IMAGE_ASSET_ENTRY_XML	isImageMediaSearch	image

- All accepted custom clipboard formats are registered formats (see MSDN Registered Clipboard Formats), which are text formats with the same characteristics as CF_UNICODETEXT.
- The actual content must be an **Asset Entry XML**, describing a video, or an image asset, respectively.
- The Asset Entry media type extends the Atom Entry media type defined by RFC 4287 and RFC 5023, by defining additional markup and additional processing rules. An Asset Entry has the Internet Media Type application/atom+xml with the RECOMMENDED type parameter type=entry.

> ⚠ **Note:** Pilot Data Server Documentation should be consulted for the Asset Entry media types. Assuming the Pilot Data Server is running on localhost, browse to http://localhost:8177/help and http://localhost:8177/AssetSearchDocumentation/.

16.4.2   API Integration

The only Windows data transfer types accepted as input are enumeration values **TYMED_HGLOBAL** with **DVASPECT_CONTENT**. The accepted **DROPEFFECT**s are *DROPEFFECT_COPY* and *DROPEFFECT_LINK*.

Links to MSDN definitions for the required enumeration values and constants are included below. Note that these links could change over time.

- TYMED_HGLOBAL
- DVASPECT_CONTENT
- DROPEFFECT

> ✅ **Tip:** If there is an error while *processing* dropped content an error message will be logged and the **Errors** button will appear if it wasn't already visible. If this happens during development, check the error log since a trace of why the dropped resource was rejected will be logged. If the media container does not accept a drop, it will likely be because the resource type is not supported; this will be indicated by the cursor displaying a stop sign.

> ✅ **Tip:** Tools such as ClipSpy can be valuable while developing the integration. To see what datatypes are provided by the drag-and-drop operation, see http://www.codeproject.com/Articles/168/ClipSpy.

16.5 Port Numbers

A summary of the required ports and connections is listed below. For detailed information, please contact Customer Support.

The table below lists all default server and listening port numbers that are used.

> ⚠ **Note:** It is recommended to run the system on a network without a firewall.

16.5.1 Listening Port Numbers

Listener	Ports	Descriptions and Comments
Oracle database	1521	For clients that connect to a Viz Pilot database running on Oracle.
Graphic Hub REST	1938	Graphic Hub REST default port. For clients that connect to a Viz Pilot database running on a Graphic Hub.
Script Runner	1981	Used to communicate with the Update Service.
Viz Pilot News	6220	Used by the Graphics Plugin for NLE to establish a connection to Viz Pilot News.
Viz Pilot	6484	Socket connection used for controlling Director using macro commands.
Viz Preview License server	7452	For Viz Pilot News using an unlicensed Viz Engine for local preview with a connection to the Viz Preview License server (this is a different server than the Preview Server).
Pilot Data Server	8177	Used when connecting over HTTP using the REST interface.
Preview Server	21098	Used when connecting over HTTP using the REST interface.

Listener	Ports	Descriptions and Comments
Viz Engine	6100 6700 6800 14300	Used by Media Sequencers that connect to a Viz Engine program and/or preview channel. Viz Engine's default program and preview port is 6100. In a single channel setup where both program and preview output is on the same machine, the default preview port is set to 6800 to separate the program and preview channels. In a dual channel setup, the default program ports are 6100 and 6800 for channel 1 and channel 2, respectively. In a dual channel setup, when used for stereo production, the default program ports are 6700 and 6800 for channel 1 (left eye) and channel 2 (right eye), respectively. An alternative port is used to avoid conflicts with port 6100 (for example when using Viz Multiplexer). Port 6100 is normally used by renderers that are on air - it is therefore recommended to use another port. Port 14300 is an optional port. The default 6100 may also be used if the renderer is not used on air.

16.6   Working Without A Database Connection

Certain features of the Viz Pilot system can be used without a database connection.

16.6.1   Viz Pilot News

To run Viz Pilot News without a database connection (NODB mode), set the following registry setting with type DWORD (32-bit) Value:

```
HKEY_LOCAL_MACHINE\SOFTWARE\Wow6432Node\[vizrt]\ActiveX x.y\TemplateFiller\NODB = 1
```

Only the Media tab will be enabled and media files will appear for all the asset search providers.

16.6.2   Director

When running Director in NODB mode, you can work with the following functionalities:

- The Media tab is available, however media objects cannot be searched.
- Playlists can be opened and run.
- Template Wizard can be opened, and templates can be created and saved.
- Object Store can be opened and you can perform image or person searches. A person created with an image can be saved. A person without an image can also be saved, but will not appear in a person search in Director.
- The following tabs are not available: Templates, Data, Viz, Files.
  To set Director in NODB mode, add a -nodb flag at the end of the target path in the shortcut properties (see Command Line Startup Options).

- Windows 64-bit: `"%ProgramFiles(x86)%\vizrt\Viz Pilot x.y\VizContentPilot.exe" -nodb`

16.6.3   Template Wizard

Use Template Wizard without a database connection. This can be useful when creating file-based templates, for example, for on-the-road template work, and template user interface design for Viz Ticker and Viz Trio

To set Template Wizard in NODB mode, set the following fields in the [TEMPLATE_WIZARD] section of ContentPilot.ini:

```
use_picturedb=n
use_database=n
```

16.7   Legacy Playlist

> ❌ **IMPORTANT!** In Viz Pilot 6.0 and later, the new Playlist Window is available. The playlist described in this section is still available, but its feature set slightly diverges from the new one.

The old playlist can be enabled in the Playlist - General Preferences options.
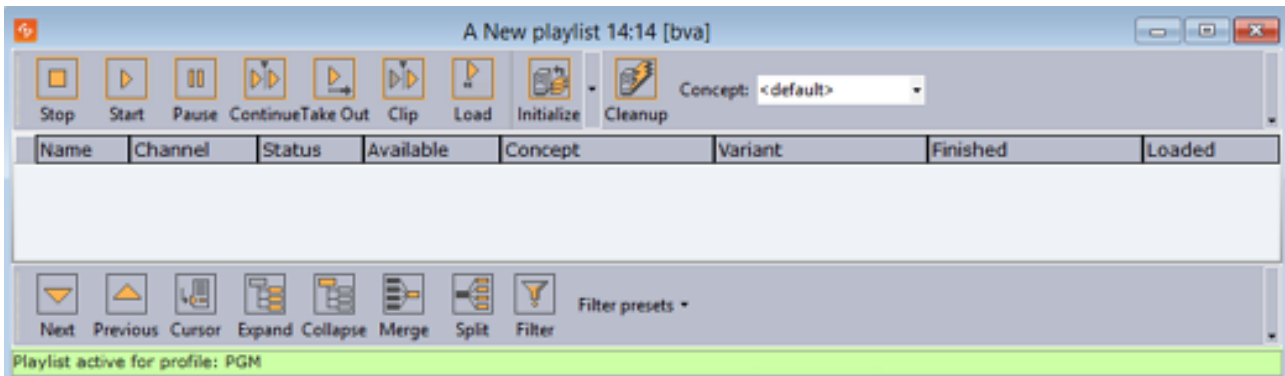
This page covers the following topics:

- Toolbars
- Playlist Overview
- Columns
- Concept
- Time Editor
- Status Bars
- Profile Bar
- Rundown Monitor Bar

16.7.1   Toolbars

There are several toolbars available which can be displayed or hidden using the Toolbar Context Menu.

Toolbar Context Menu

Right-clicking the toolbar gives you access to the context menu with the following options:

- **Customize Toolbar:** Opens the Button Configuration dialog box. For details about this dialog box, see Custom Toolbar.
- **Item Toolbar:** Displays the Item Toolbar (enabled by default).
- **Playlist Toolbar:** Displays the Toolbar (enabled by default).
- **Video Toolbar:** Displays the Video Toolbar (disabled by default).
- **Custom Toolbar:** Displays the Custom Toolbar (disabled by default).

Item Toolbar

The Item toolbar is located at the top of the playlist window:
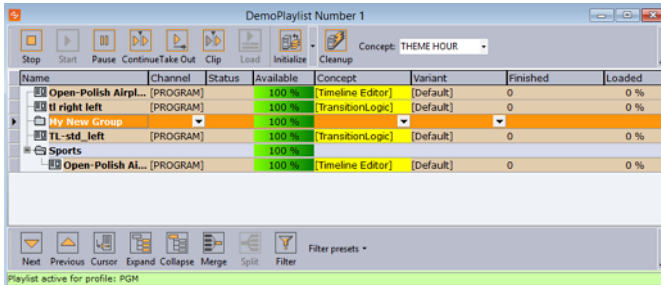
- **Stop:** Cancels a playlist item that has been scheduled to run. Stop is only relevant when groups with elements with timecode set have been triggered.
- **Start:** Runs the scene from the current time frame.
- **Pause:** Pause the scene at the current time frame. Pause is only relevant for clip playback.
- **Continue:** Continues to play the scene from the current time frame. To be able to use the Continue function, the scene must either have been affected by the Pause function or have a stop point.
- **Take Out:** Clears a scene based item, and takes a transition logic based scene to the out state stop point.
- **Clip:** Ignores the animation and takes the scene to a defined Clip point. A Clip point is defined by adding a tag named *noanim*. To learn more about adding Clip points, see the Tag points section of the *Viz Artist User Guide*.
- **Load:** Loads the scene into Viz Engine.
- **Initialize:** Initializes or re-initializes the entire playlist. Scenes are loaded in memory for the channel that is defined per data element (see the channel column in the playlist). A status indicator at the bottom of the playlist is displayed while initialization is in progress.
    - **Initialize Playlist:** Initializes or re-initializes the entire playlist (default).
    - **Initialize Selected:** Initializes the selected items in the playlist.
- **Cleanup:** Cleans up all renderers that have an item in the current playlist (clears all loaded graphics from memory for those channels). To clear all renderers, go to the Main Menu > Viz Engine Menu > **Cleanup Renderers**
- **Concept:** Sets a concept for the whole playlist. If the value is set to *default*, all data elements will use the concept that is set in the Concept column. See also Concept.
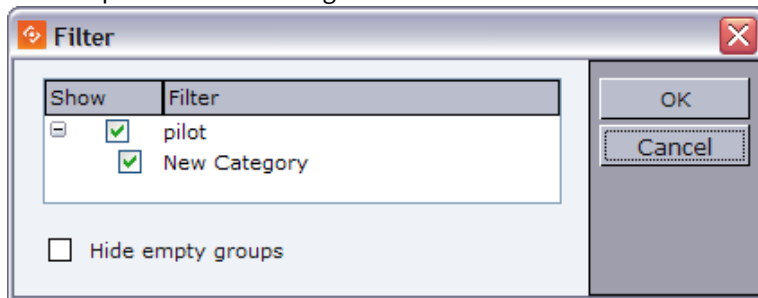
Toolbar

The Playlist Toolbar is located at the bottom of the window:

- **Next:** Selects the Item cursor (black pointer) to the next element in the playlist.
- **Previous:** Selects the Item cursor to the previous element in the playlist.

- **Cursor:** Jumps the Item cursor to the location of the External GPI Cursor (yellow arrow).
- **Expand:** Expands the groups in the playlist.
- **Collapse:** Collapses the groups in the playlist.
- **Merge:** Merges the current folder, so that the subfolders become unavailable.
- **Split:** Splits a folder that is merged.
- **Filter:** Opens the Filter dialog box.

The Filter window is used to enable or disable playlist filters. Each playlist can enable or disable different filters. Filters are based on how templates are categorized in Template Wizard.

Video Toolbar

The Video Toolbar is not displayed by default, and is only available for MVCP integrations.

When selected, the Video Toolbar is by default positioned below the Playlist Toolbar.

> ⚠ **Note:** The Video Toolbar settings only work if the system is configured to be integrated with a video server.

On Air: Sets the playlist in On Air mode. When the playlist is set to On Air an On Air icon ‖‖‖‖ appears in the MSE Playlists list.

- **Allocate:** Automatic channel allocation starts, the first clips are cued up, and the External GPI cursor is set to the first story that contains a video clip.
- **Off Air:** Sets the playlist in Off Air mode.
- **Auto Cue:**
  - **Enable Auto Cue:** Enables the automatic channel allocation.
 - **Disable Auto Cue:** Disables the automatic channel allocation.

Example

If a playlist uses two channels (A and B) and the playlist is set in On Air mode, the first clips allocated to these channels will be cued. Channel A is taken On Air by a vision mixer GPI (advance and take). This also takes the studio tally (red light on monitors) high. This is received by the Media Sequencer as a GPI, and the Timecode Monitor indicates high tally on that channel. The vision mixer cuts to camera or to another video source. This takes tally low

on that channel, and Media Sequencer receives a GPI low signal. After 5 seconds, the tally goes low, and the next clip allocated to that channel will be cued.
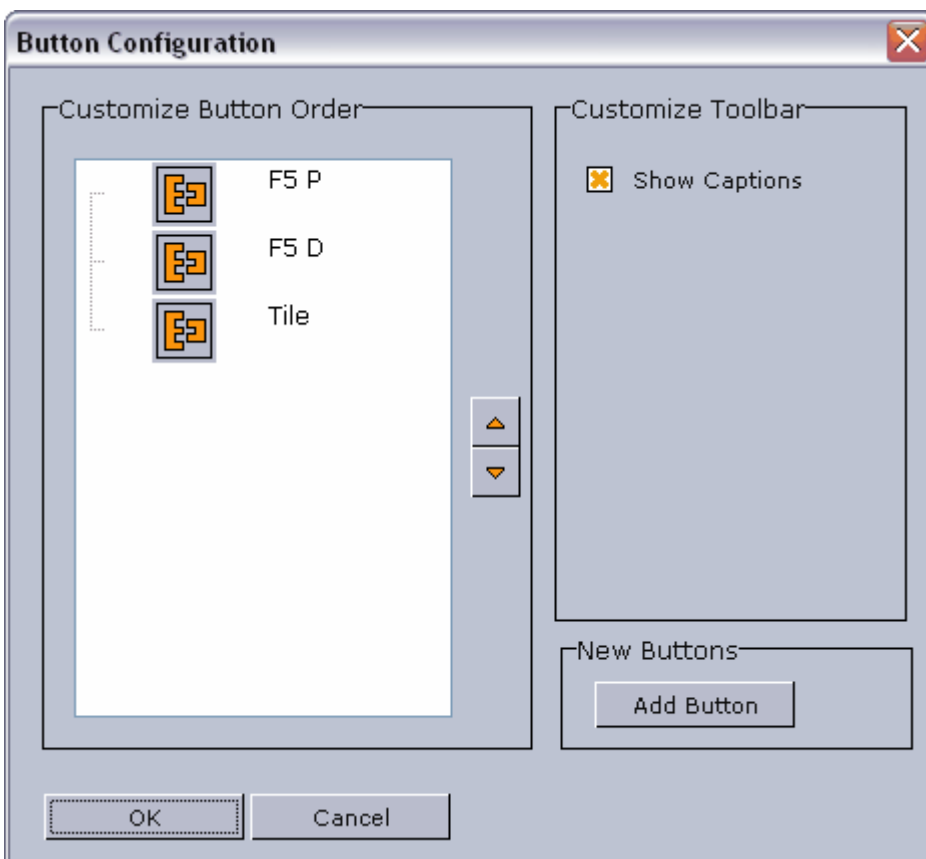
Custom Toolbar

The Custom Toolbar is not selected by default. When selected, the Custom toolbar is positioned above the Playlist toolbar.
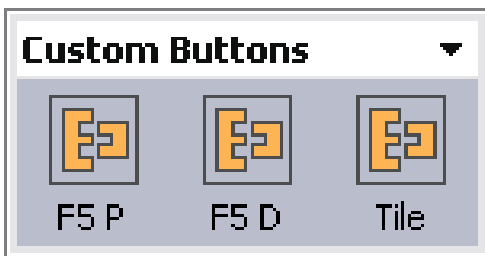
Configuring Buttons

Selecting Customize Toolbar from the Toolbar Context Menu will open the Button Configuration window:

The customized buttons list is empty by default. If the list contains two or more custom buttons, the order of the buttons can be rearranged with the Up and Down buttons.

If **Show Captions** is enabled, the buttons in all buttons in all toolbars will be displayed with their respective names.

Add Command

Clicking **Add Button** in the Button Configuration window will open the Add Command window:

- **Description:** Add a descriptive text to the new command button.
- **Image:** Add an image or icon to visualize the new command button.
- **Simple button:** Add a predefined function by clicking the ellipsis (**...**) button to use macro commands or Viz Engine commands (see image below).
    - **Command:** Click the ellipsis (**...**) button to open the Predefined Functions window. Select a function to add to the button.
- **MSE Status button:** Depending on the status of a button ("on" or "off"), it can have different behaviors and trigger different functions. An MSE button is defined to monitor the Media Sequencer.
    - **Subnode of:** Root, Profile or Playlist.
    - **MSE nodepath:** Sets the relative path of the sub node.
    - **Value:** Any given value in the Media Sequencer.
    - **Exists:** When checked the button is turned "on" if the node exists. Independent of the value field.
    - **While down:** Command that is triggered while button is down.
    - **While up:** Command that is triggered while button is up.
      Both the Simple and the MSE buttons can use Predefined Functions. The functions can be user defined or built-in functions.

Example

- **Example 1:** The button is "on" if the path "*/hello"* in the Media Sequencer has the value "*aaa*"
  - Subnode of = Root
  - MSE nodepath="hello"
  - Value="aaa"
- **Example 2:** The button is "on" if the path "*/config/profiles/Studio/top/sub"* in the Media Sequencer has the value "*aaa"*.
  - Subnode of = Profile
  - MSE nodepath = "/top/sub"
 - Value = "aaa"

Creating a Custom Button

1. Right-click a toolbar and select **Customize Toolbar**.
2. In the Button Configuration window click the **Add Button** button.
3. In the Legacy Playlist#Add Command window enter a description for the button in the **Description** text field.
4. Select the Simple button option, and click the ellipse (**...**) button to add Predefined Functions.
5. Click **OK**.

> ⚠ **Note:** When adding button images, use BMP images of approximately 30 x 40 pixels.

16.7.2   Playlist Overview

A playlist can contain various groups and elements. Each group and element is presented as a row in the Playlist.

Cursors
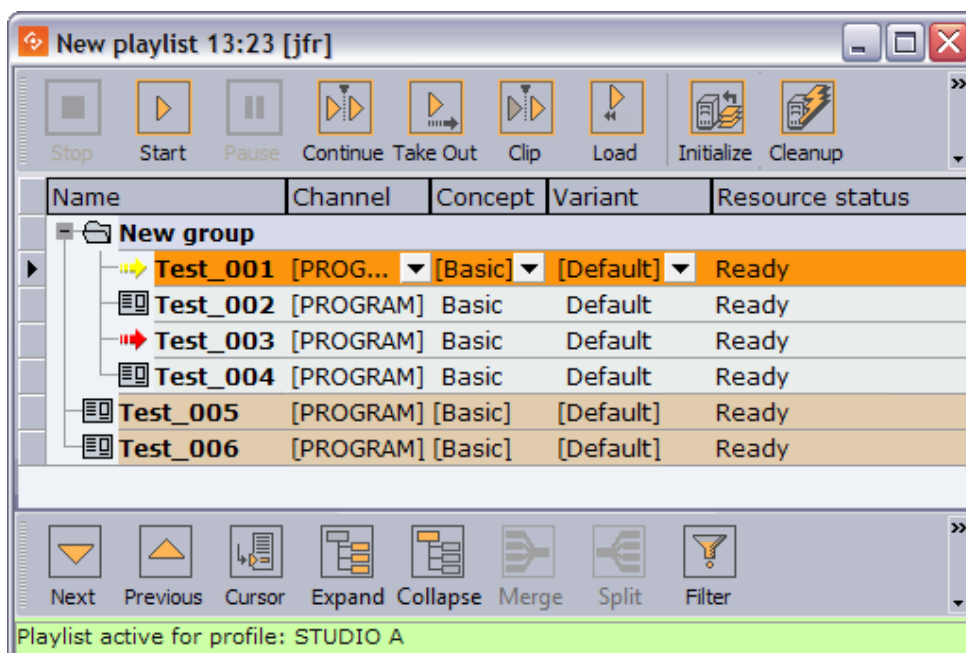
There are three cursors in the Director playlist:

Current Item Cursor

- Red arrow ⏩, this is the **current** selected element.
- Indicates that this is the next element that will be taken on-air.

External GPI Cursor

- Yellow arrow ⏩ the **next** element that will be focused (selected) when the currently selected element is taken. Referring to the screenshot above: When *Test_003* is taken, *Test_001* will then be the next current element and displayed with a red cursor.
- When GPI is enabled, the external GPI system's cursor will be displayed in any client that is using the same profile as the external system.
- The Cursor options in the Legacy Playlist Context Menu apply to this cursor.

Director Cursor

- Black pointer in the grey sidebar ▶. This is the orange highlighted row.
- Shows the item that has been manually selected in the Director playlist.
- The Next, Previous and Cursor buttons in the Toolbar apply to this cursor.

Context Menu
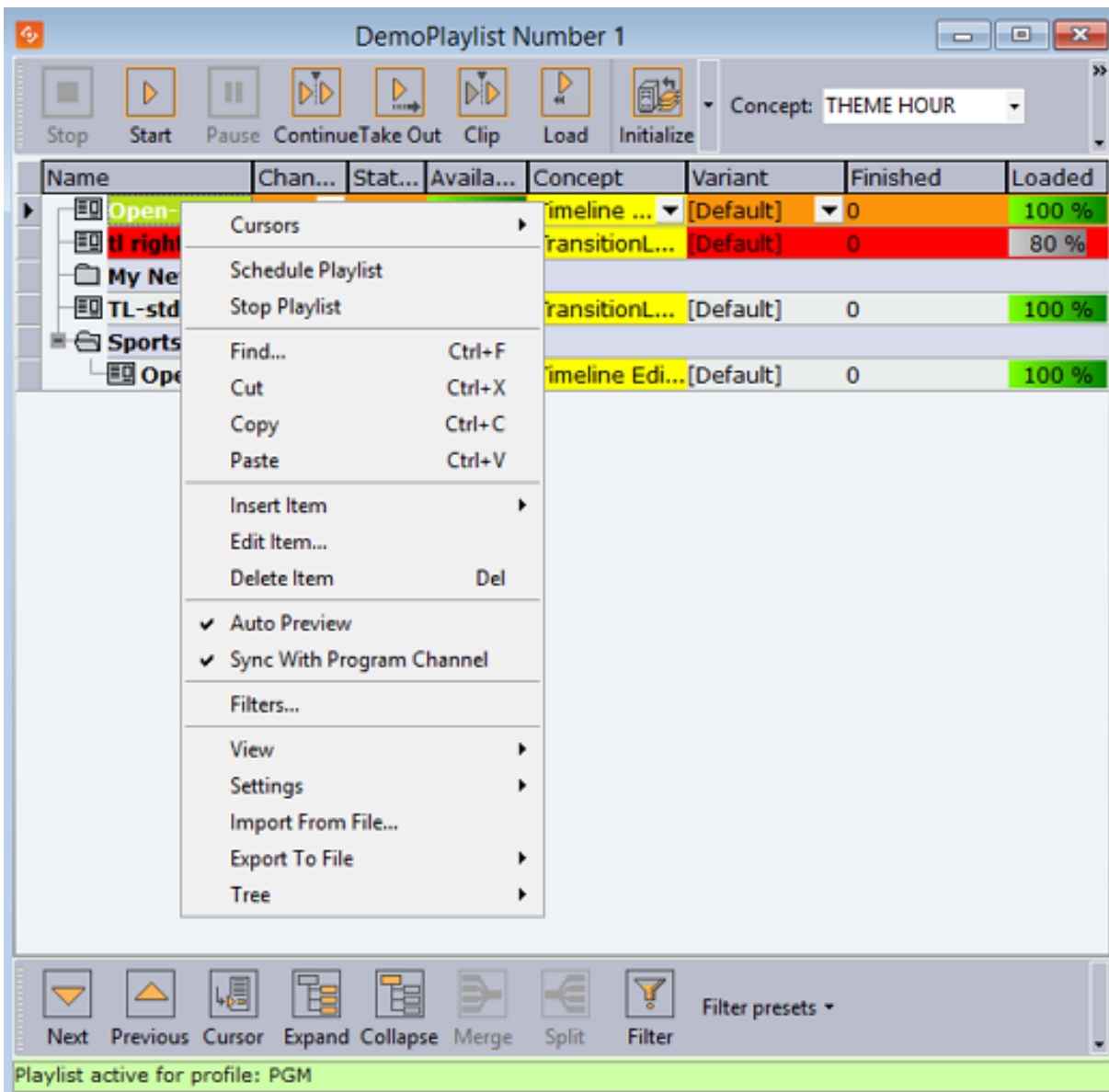
- **Cursors:** Shows three options for setting, clearing and resetting the External GPI Cursor (yellow arrow).
    - **Reset Cursor:** Sets the external cursor back to the point it was before a Set Next was executed.
    - **Set Next:** Sets the external cursor for a selected element. The cursor indicates the element that will be taken next if, for example, a "take and advance" General Purpose Input (GPI Tab) action is executed.
    - **Clear Next:** Clears the Set Next cursor. This option is only available when clearing a cursor set with the Set Next option.
- **Schedule Playlist:** Activates the playlist. If for example Begin time has been set, the playlist must be scheduled before the changes take effect.
- **Stop Playlist:** Stops a previously scheduled playlist.

- **Find:** Shows the Playlist Search area below the Playlist.

Typing the first letters of the name of a playlist element in the Find text box, marks the closest element in the playlist (either above or below the current element, depending on whether the Find Next or Find Previous button is clicked) that corresponds to the letters in the text box. The Find Next button is selected by default. Clicking the X button hides the Playlist Search area.
- **Cut:** Removes the current element from the playlist.
- **Copy:** Copies the current element in the playlist.
- **Paste:** Adds a copied element above the currently selected element in the playlist.
- **Insert Item**
    - **Group:** Adds a new group to the playlist, above the currently selected element. To place an element into the group, press the CTRL key, and then drag and drop the element onto the desired group. The action is indicated by an angled arrow. Groups can also be organized as sub-groups.
- **Edit Item:** Displays the editable text elements in the item editor. Allows last minute changes to text in graphics.
- **Delete Item:** Deletes the current Playlist element. It is only removed from the playlist.
- **Auto Preview:** Automatically displays selected playlist elements using the preview channel.
- **Sync With Program Channel:** This is an option for the Auto Preview feature available above. If Auto Preview is on, this setting affects what is shown on the preview channel when a user selects a new element.
    - If **on**, the current state of the preview channel is synchronized with the program channel before the selected element is taken on it. This mainly affects transition logic scenes as this means that the selected element will be shown in-context with the same elements on-screen as in the program channel.
    - If **off**, the preview channel will be cleared before the current element is taken. For example, this means that a transition logic element will be shown alone on the preview screen regardless of what is shown currently on the program channel.

    > ⚠ **Note:** Sync With Program Channel is permanently enabled in situations where the Media Sequencer does not support clearing before a take.

- **Filters:** Opens the Filter dialog box, where types of elements that should be hidden in the playlist can be selected.
- **View**
    - **Visible Toolbars:** Selects which toolbars are visible (Item Toolbar, Playlist Toolbar, Custom Toolbar).
    - **Show Item Editor:** Shows the Item editor at the right side of the Playlist list. The Item editor displays editable text for the selected graphics item in the playlist. The editor can be expanded, hidden or closed by clicking the arrow and X buttons in the upper right corner.
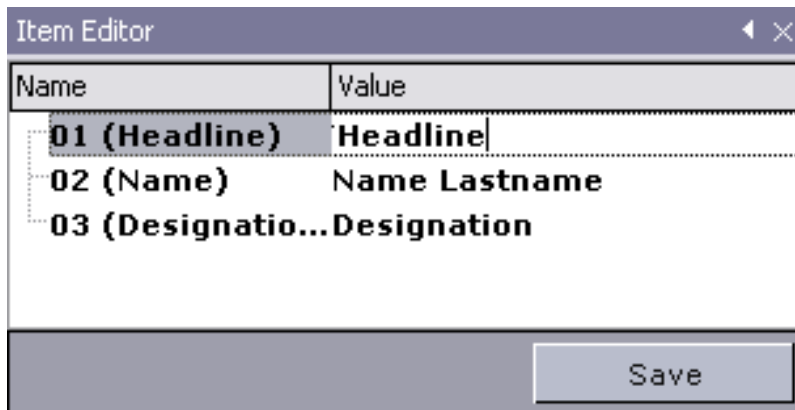
- **Show Image Bar:** Shows a bar with thumbnails of the elements in the playlist. To hide the Image bar, click the X button in the upper left corner. Right-clicking the Image bar opens the Change Image Size dialog box.
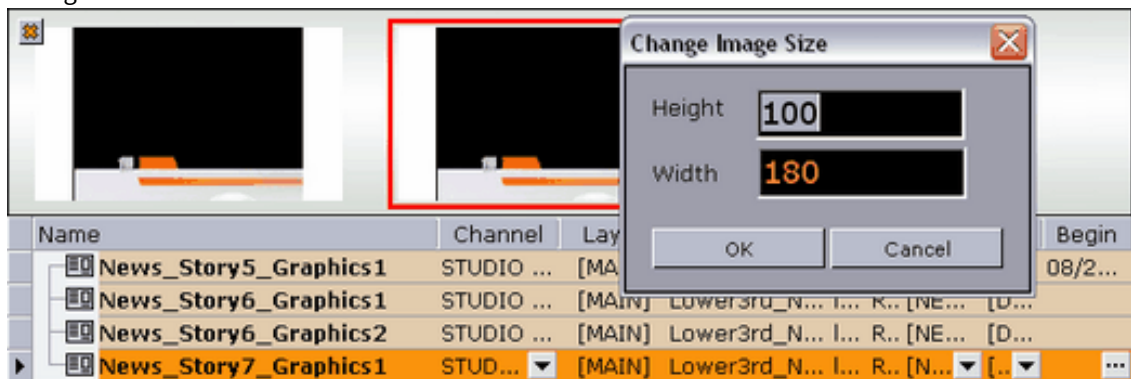
- **Auto Width:** When enabled, the columns are minimized, so that they fit perfectly to the content. This setting stores the setting for the current playlist. See also the Store as Default option.
- **Font:** Opens a dialog for setting font, font style, size and so on. The font settings change the text in the Playlist list.
- **Image Size:** Sets the thumbnail images to the selected pixel size.
- **Settings**
  - **Store as Default:** Sets the current settings (selected columns, column widths, filters, etc.) as default for all new playlists. Remember to disable Auto Width for this function to work properly.
  - **Restore Default:** Restores the settings to the original setup.
- **Import From File:** Opens the Open dialog box, where a previously exported XML file can be searched for and imported.

⚠️ **Note:** The Import from File does not accept characters outside the Basic Multilingual Plane (BMP). If a non-BMP character is found, the import will stop and return an error message stating the position of the first non-BMP character.

- **Export To File:**
  - **Playlist:** Opens the Save As dialog box, where the entire playlist can be exported.
  - **Selected Items:** Opens the Save As dialog box, where the selected elements in the playlist can be exported.

- **Tree**
    - **Select All:** Selects all the items in the playlist.
    - **Expand All:** Expands all groups in the playlist.
    - **Collapse All:** Collapses all groups in the playlist.

Grouping Elements in a Playlist

1. Right-click the Playlist and select **Insert Item** -> **Group** from the Context Menu.
2. Place an element in a group by holding the **CTRL** key down while performing a drag and drop operation. An angled arrow indicates that the element may be dropped.

16.7.3   Columns

Playlist columns can be displayed or hidden by using the Column Context Menu. Dragging and dropping the column headings changes the order of the columns.

Many of the columns are presented by default. These are; Name, Channel, Status, Available, Concept, Variant, Finished and Loaded.

> ✅ **Tip: F2** can be used to change the value of a data element's column setting.

Column Context Menu

The context menu which appears when right-clicking the column headings includes:

- **Video:** Displays the following four video options:
    - **Available:** Displays the status of external resources needed by the Viz Engine (e.g. transferred video, and if it is available on the video playout engines). This column was previously named Progress.

    - **Mark In:** Sets mark in times for video clips.
    - **Mark Out:** Sets mark out times for video clips.
    - **Postroll:** Sets postroll for video clips. When a video clip is played, the time and remaining time are shown in the Timecode Monitor. The remaining time; zero at clip-end if postroll is set to 0. Normally it is desirable with a few seconds of postroll so that the vision mixer can cut to another source without risking displaying a freeze frame. With a postroll of 3, the countdown/remaining time will be zero 3 seconds before clip end and count from zero and downwards for three seconds.
- **Misc:** Displays miscellaneous options. Begin, end, timecode and duration can be set directly in the playlist. Some newsroom systems also propagate timing information to their control applications for monitoring and playout of newsroom playlists. Such information can for example be sent using the MOS protocol.

> ⚠️ **Note:** Director is able to override timing information from newsroom playlists, however, this is not recommended.

- **Begin:** Shows the start time of a group. Hence, begin and end times can only be set for groups and not elements of a group. To set time for elements, use the Timecode and Duration columns. See also the Time Editor section.
- **Category:** Shows the category the Sub Category resides under. Default category for Template Wizard templates is the *pilot* category.
- **Channel:** Shows which output channel an element should be sent to. Various elements can be sent to different output channels. The output channels can be set directly in the column. By default the main (program) output channel is selected, but this can be overruled by setting an alternative channel in a template (that all data elements made from it will have). This column is presented by default.
- **Duration:** Sets the length of the element. This time is relative to the Begin and Timecode parameters.
- **End:** Shows the end time for a group. See also the Time Editor section.
- **External Id:** The External Id can be set directly in the column. It attaches a data element execution to a specific external signal that comes from a button box or similar device. An External Id is used with the Quick CG feature for adding embedded MOS objects to a newsroom story.
- **Finished:** If a data element is checked off as finished when saved it can be shown in the Finished column in the playlist displaying either 1 for finished and 0 for not finished.
- **Loop:** Loops a fullscreen video clip on the video channel. Default behavior is to play once.
- **MOS ObjAir:** Marks if the newsroom element is marked as ready. The element can only be taken on air if it is checked (ready). Using the macros `playlist:next_ready` and `playlist:previous_ready` will skip those elements that are not ready. Do not use the macros `playlist:next` or `playlist:previous`.
- **Name:** Shows the element name as it was saved in the database. This is presented by default.
- **Resource Status:** Indicates if an element is ready for playout. It will also show an error status if the element was not loaded or made available.
- **Status:** Shows the status for video clips; cuing, cued, playing, paused, or aborted.
- **Story #:** Shows the name or number of the Story as used in the NCS. This is an optional field originally intended for use by prompters.
- **Sub Category:** Shows the category a template is assigned to. See Categories and Channels in Template Wizard.
- **Thumbnail:** Shows thumbnails of the scenes.
- **Timecode:** Sets the timecode for the element. This setting affects both graphics and video elements. The timecode can be set directly in the column. To trigger elements based on timecode, the elements must be in a group, and the group must be executed. An element's timecode must be set within the group's Begin and End time.

> ⚠️ **Note:** To monitor video elements in graphics using the TimeCode monitor, the Viz Engine that renders graphics must also be defined in the profile configurations for video playout.

- **Graphics:** Displays the graphics options.

- **Concept:** Shows which concept(s) the data element is associated with.
- **Effect:** Displays transition effects that can be set per data element. When the effect is executed it will have an effect on the data element it was set for, and the next data element taken. See also how To enable transition effects.
- **Effect Image:** Displays an image of the transition effect scene.
- **Layer:** Allows loading of graphics in separate layers on Viz Engine (front, middle, back). For example, a lower third can be shown in front of a virtual studio set or any other background, or a bug can be shown in the front layer while a lower third is shown in the middle layer. This column is presented by default.
- **Loaded:** Shows the loaded status (in memory) of the scene and images used for a data element of that scene.
- **Template:** Shows the name of the template. This column is presented by default.
- **Variant:** Select a concept's variant from the drop-list (see the Concept column).
- **Index:** Sets an index number for each row in the playlist. This number will always be sorted either ascending (lowest to highest). If a row is moved or deleted the index changes accordingly.
- **Auto Width:** When enabled (selected), Auto Width expands the columns and distributes them evenly.

Enabling Transition Effects

1. Configure a program channel that has a Viz in On-air mode.
2. Make sure the scene effects folder in Viz is named **transitions** and placed at the root level of the scene directory.

Adding a Transition Effect

1. Right-click the playlist's column headers, and select Graphics, and then Effect and Effect Icon (last is only for visual reference)
2. Click the arrow button in the Effect column and select, from the drop list, the transition effect.

16.7.4   Concept

In most cases Director playlists are created in Newsroom Computer Systems (NCS). A playlist created using an NCS client, can consist of data elements within a specific concept. For example a News or Sports concept.
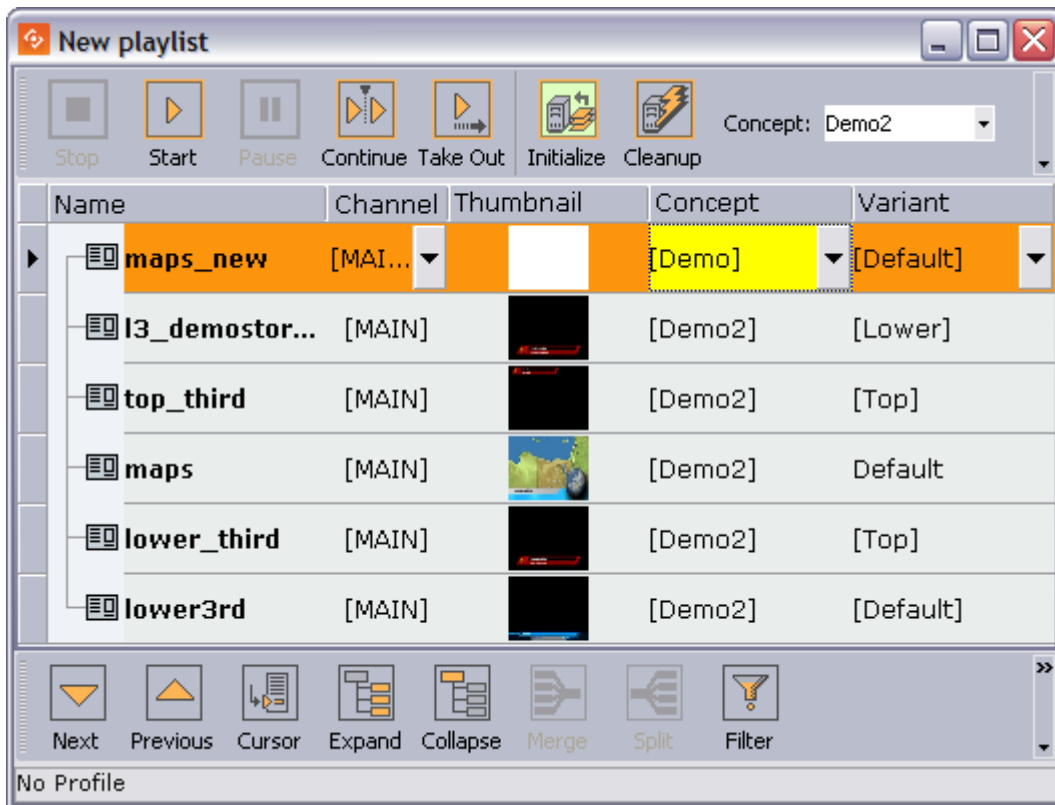
Director is able to override a playlist's concept. Data elements, saved to a specific concept, can belong to several concepts. Thus, a data element that does not belong to the overridden concept is highlighted in yellow.

Some newsroom systems support the use of master concepts (for example ENPS) set by the newsroom system itself. A master concept is able to override a concept set when creating a data element.

When a concept is overridden, the playlist will, when opened in Director for monitoring and playout, show all data elements with the concept they were saved with, but played out using the Master Concept. Hence, if a data element was saved with a Sports concept, and the master concept is News, the News concept is used.

Concept:	Customers		

Template	Concept	Variant	Resource status
	▼	▼	
Lower3rd	B Sports	Lower	Ready
Visual Data Tools	[B Sports]	[Bars]	Ready
Visual Data Tools	[A News]	[Line]	Ready

If the Master Concept is changed, and the data elements are not part of the master concept, the data element's concept column will be highlighted with a yellow color.

If a data element's concept is manually changed in Director, the master concept will not override the new concept, and a warning is not issued since the concept is manually overridden.

The concept a data element is saved with is depicted with square brackets* **[ ]***. A data element's concept and variant that are manually changed in Director do not show the concept and variant name with square brackets.
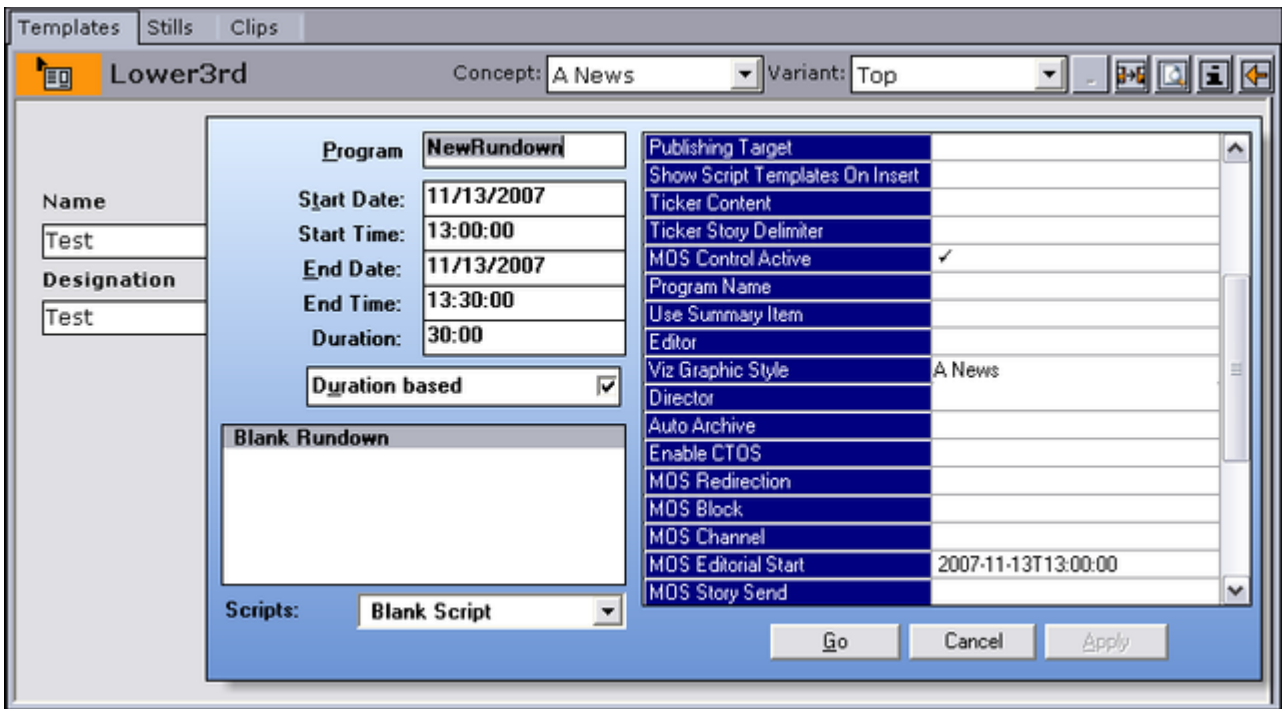
MOS Graphics Concept

The concept implementation of the MOS protocol, that is supported by Director and Viz Gateway, can be used by any newsroom system that use the MOS protocol. However, the feature must be implemented.

Below is an excerpt of a MOS message sent from the newsroom system when a playlist is created with a master concept.

```xml
<mosExternalMetadata>
    <mosScope>PLAYLIST</mosScope>
...
     <mosPayload>
 ...
        <GraphicConcept>A News</GraphicConcept>
...
    </mosPayload>
</mosExternalMetadata>
```

In ENPS it is possible to set the concept (or style) when creating a rundown (playlist). Set the rundown property *Viz Graphics Style* to the same as the graphics concept used.

> ⚠️ **Note:** The naming convention must be the same for the scene design and the playlist created in the newsroom system.

16.7.5   Time Editor

The Time Editor is displayed when clicking the ellipsis (**...**) button in the Begin or End columns:
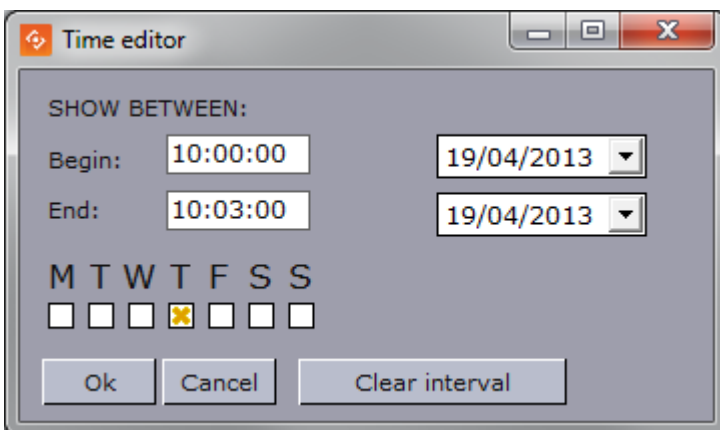
In this window, a group can be set to run between a start and stop date and time. It can also be set to run on specific weekdays. Once a start and stop time have been set, the date and time is shown in the Begin and End column.

16.7.6   Status Bars

The playlist displays two types of status bars at the bottom of the window, depending on the type of playlist that is open (Director playlist or MOS playlist). These two status bars operate independently of each other.

The Profile Bar is available for all playlists, while the Rundown Monitor Bar is only available for MOS playlists.

16.7.7   Profile Bar

The Profile Bar is available for all playlists. It shows whether the playlist is active or inactive, and if active, the name of the profile it is active in.

This information is stored in the Media Sequencer, so can be accessed by all Directors that are connected on the same profile.
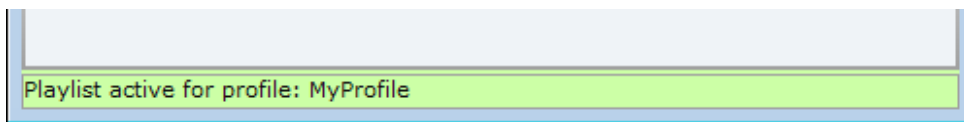
Context Menu - Profile Bar

When a playlist is opened, it retains the active/inactive status it had before it was closed. The context menu can be used to change the status.

- **Activate in current profile:** Activates and monitors the selected playlist in the current profile, and enables the Media Sequencer to trigger transfer of video clips to all Viz Engines in the current profile.
- **Deactivate:** Deactivates and stops monitoring the playlist.

> ⚠️ **Note:** If the active profile is changed or renamed, the playlist must be reactivated in the new profile by deactivating and then activating the playlist.

Cleanup and Auto Preview work on playlists that are either active or inactive. Initialize will only work on active playlists.

16.7.8   Rundown Monitor Bar

The Rundown Monitor Bar is available for MOS playlists (also known as newsroom playlists).
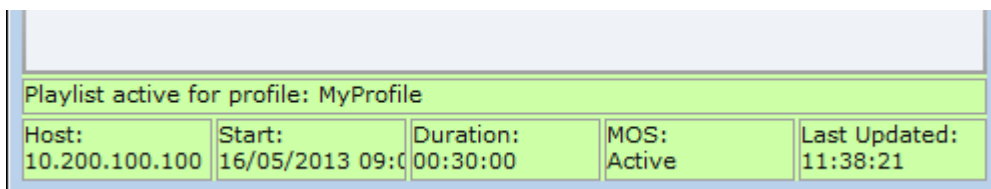
This bar is used in Vizrt's MOS integration, and shows the current MOS status for the playlist.

When a MOS playlist is first opened in Director, it is empty. Monitoring needs to be requested using the context menu so that the playlist can be populated.

Once Director requests the monitoring of a MOS playlist, it can be presented as a playlist in Director. Director can then monitor the playlist and receive updates from the newsroom system and Director. The operator can then take the graphics on air from Director.

Status Information

The information available in the rundown monitor bar is as follows:

- **Host:** IP address of the Media Sequencer that has requested the playlist.
- **Start:** Playlist start time, according to the Newsroom system.
- **Duration:** Playlist duration, according to the Newsroom system.
- **MOS:** The current MOS status for the playlist (Active/Inactive).

Context Menu - Rundown Monitor Bar

- **Request:** Begin monitoring, and populate the Director playlist with the elements from the newsroom system. Once the monitoring is active, all subsequent updates done to the newsroom rundown will be reflected in the **active Director playlist**.
- **Release:** Director will stop monitoring the newsroom system. A list of elements will still appear in the Director playlist, but any subsequent updates to the Newsroom rundown will not be reflected in the **inactive Director playlist**.

> ❌ **IMPORTANT!** Any elements that are manually added to the playlist by Director will disappear if a released playlist is requested again.

> ⚠️ **Note:** If using Director to take items on air, be mindful that the cursor position in Director may change if the related item is deleted from the newsroom rundown. The **Current Element** cursor would move to the following item, and the **Next Element** cursor would be removed.

Cleanup and **Auto Preview** will work on playlists that are either active or inactive. Initialize will work once the MOS playlist has been populated with content.

Example Scenarios

If two Directors use the same Media Sequencer and MOS Gateway and have the same playlist Requested, they will both receive all updates.

If two Directors use the same MOS Gateway but different Media Sequencers, and they both activate the same MOS rundown, the last Director to activate the rundown will receive any MOS updates that come from the Newsroom system.

See Also

- Using Categories and Channels in Template Wizard.
- Profile Status Bars
- To create a custom button

16.8   Log Files

This section contains the following topics:

- Viz Log Files
- Crash Log Files
- Viz Config Files
- Modifying Log File Settings

16.8.1   Viz Log Files

Viz Pilot System Log Files

The default location is:

- `%PROGRAMDATA%\Vizrt\Logs\Viz Pilot x.y\pilot.log`

> ⚠ **Note:** If this path is empty, check the ContentPilot.ini, as the default path may be overridden by the logdir setting in the **Pilot** section of the ini file.

You can also access the logs from **Main Menu > File >** View Log.

The logging level can be set using environment variable PILOT_LOG_LEVEL. This accepts the values 0-4 where a higher number means a more verbose log. The default level is 3.

Database Log Files

See Database Error Logs.

Other Log Files

Other useful log files include the Media Sequencer logs and the Viz Engine logs.

16.8.2   Crash Log Files

Eureka Log Files (.elf)

Eureka Log files for Viz Pilot News, Director and Template Wizard are found here:

- `%ProgramData%\Vizrt\Logs\Viz Pilot x.y`

Crash Dumps

When using Windows 7 or later, it's possible to generate crash dumps automatically by setting certain registry keys, as described in Collecting User-Mode Dumps.

All of the Viz Pilot system's client applications are 32-bit. When collecting a crash dump of a 32-bit application on 64-bit Windows, you must use the 32-bit version of the Windows Task Manager in order for the dumps to be readable.

The Windows task manager will normally be installed in:

- 32-bit version; C:\Windows\SysWOW64\taskmgr.exe
- 64-bit version; C:\Windows\System32\taskmgr.exe

16.8.3   Viz Config Files

Viz Pilot System Config Files

Most of the Viz Pilot system's components use the `ContentPilot.ini` file. Object Store may use its own `objectstore.ini` file.

The INI files are stored here:

- `%ProgramData%\Vizrt\Viz Pilot x.y\Config`
  For more details, see Initialization and Configuration Files.

Viz Engine Config Files

The Viz Engine files are stored here:

- Windows 64-bit: `%ProgramFiles(x86)%\Vizrt\Viz3`

Media Sequencer Config Files

The `default.xml` files are stored here:

- Windows Vista and 7: `%ProgramData%\Vizrt\Media Sequencer Engine`
- Windows XP: `C:\Documents and Settings\All Users\Application Data\Vizrt\Media Sequencer Engine`
  For more details, see Media Sequencer Data.

16.8.4   Modifying Log File Settings

The default location can be overridden using the LOGDIR and LOGFILE settings in the ContentPilot.ini file.

The maximum size and number of rolling log files can also be configured in the ini file (LOGSIZE, LOGCOUNT).

The log level can be changed by using the following command line parameters:

- `noverbose` sets log level to low.
- `extra_verbose` sets log level to high.